A parallel second order Cartesian method for elliptic interface problems and its application to tumor growth model

POLITECNICO DI TORINO

SCUOLA DI DOTTORATO
Dottorato di Ricerca in Fluidodinamica

UNIVERSITÉ DE BORDEAUX 1

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET INFORMATIQUE
Specialité: Mathématiques Appliquées

Tesi di Dottorato - Thése

A parallel second order Cartesian method for elliptic interface problems and its application to tumor growth model

---

Marco Cisternino

Tutori - Directeurs:
prof. Angelo Iollo
prof. Luca Zannetti

APRILE 2012

Abstract

This theses deals with a parallel Cartesian method to solve elliptic problems with complex interfaces and its application to elliptic irregular domain problems in the framework of a tumor growth model. This method is based on a finite differences scheme and is second order accurate in the whole domain. The originality of the method lies in the use of additional unknowns located on the interface, allowing to express the interface transmission conditions. The method is described and the details of its parallelization, performed with the PETSc library, are provided. Numerical validations of the method follow with comparisons to other related methods in literature. A numerical study of the parallelized method is also given. Then, the method is applied to solve elliptic irregular domain problems appearing in a three-dimensional continuous tumor growth model, the two-species Darcy model. The approach used in this application is based on the penalization of the interface transmission conditions, in order to impose homogeneous Neumann boundary conditions on the border of an irregular domain. The simulations of model are provided and they show the ability of the method to impose a good approximation of the considered boundary conditions.

Keywords: elliptic interface problem, Cartesian method, second order scheme, interface unknowns, parallel method, tumor growth, penalty method, elliptic irregular domain problem, homogeneous Neumann boundary conditions.

Abstract

Cette thèse porte sur une méthode cartésienne parallèle pour résoudre des problèmes elliptiques avec interfaces complexes et sur son application aux problèmes elliptiques en domaine irrégulier dans le cadre d'un modèle de croissance tumorale. La méthode est basée sur un schéma aux différences finies et sa précision est d'ordre deux sur tout le domaine. L'originalité de la méthode consiste en l'utilisation d'inconnues additionnelles situées sur l'interface et qui permettent d'exprimer les conditions de transmission à l'interface. La méthode est décrite et les détails sur la parallélisation, réalisée avec la bibliothéque PETSc, sont donnés. La méthode est validée et les résultats sont comparés avec ceux d'autres méthodes du même type disponibles dans la littérature. Une étude numérique de la méthode parallélisée est fournie. La méthode est appliquée aux problèmes elliptiques dans un domaine irrégulier apparaissant dans un modèle continue et tridimensionnel de croissance tumorale, le modèle à deux espèces du type Darcy . L'approche utilisée dans cette application est basée sur la pénalisation des conditions de transmission à l'interface, afin de imposer des conditions de Neumann homogènes sur le bord d'un domaine irrégulier. Les simulations du modèle sont fournies et montrent la capacité de la méthode à imposer une bonne approximation de conditions au bord considérées.

Mots clès: problème elliptique avec interface, méthode cartésienne, schéma d'ordre deux, inconnues sur l'interface, méthode parallèle, croissance tumorale, méthode de pénalisation, problème elliptique dans un domaine irrégulier, conditions aux bords de Neumann homogènes.

Abstract

Questa tesi introduce un metodo parallelo su griglia cartesiana per risolvere problemi ellittici con interfacce complesse e la sua applicazione ai problemi ellittici in dominio irregolare presenti in un modello di crescita tumorale. Il metodo è basato su uno schema alle differenze finite ed è accurato al secondo ordine su tutto il dominio di calcolo. L'originalità del metodo consiste nell'introduzione di nuove incognite sull'interfaccia, le quali permettono di esprimere le condizioni di trasmissione sull'interfaccia stessa. Il metodo viene descritto e i dettagli della sua parallelizzazione, realizzata con la libreria PETSc, sono forniti. Il metodo è validato e i risultati sono confrontati con quelli di metodi dello stesso tipo trovati in letteratura. Uno studio numerico del metodo parallelizzato è inoltre prodotto. Il metodo è applicato ai problemi ellittici in dominio irregolare che compaiono in un modello continuo e tridimensionale di crescita tumorale, il modello a due specie di tipo Darcy. L'approccio utilizzato è basato sulla penalizzazione delle condizioni di trasmissione sull'interfaccia, al fine di imporre condizioni di Neumann omogenee sul bordo di un dominio irregolare. Le simulazioni del modello sono presentate e mostrano la capacità del metodo di imporre una buona approssimazione delle condizioni al bordo considerate.

Parole chiave: problema ellittico con interfaccia, metodo cartesiano, schema al secondo ordine, incognite sull'interfaccia, metodo parallelo, crescita tumorale, metodo di penalizzazione, problema ellittico in dominio irregolare, condizioni al bordo di Neumann omogenee.

Acknoledgements

I would like to thank all the people having followed me during the three Ph.D. years. I do it without a special importance order because I don't think I would be able to make a true ranking.

However, surely, the first place is for Angelo Iollo and Luca Zannetti. They give me the possibility of working with them, proposing me the subject of this thesis. Their experience and their advices have been a very important support to my work and to my scientific growth in general. My collaboration with them allows me to enter a new scientific domain and to enhance my formation from an international point of view. This work is surely the main effect of their trust in me.

A very special thanks goes to Lisl Weynans. Working with her has been an honor and a pleasure. We developed together the method, starting from an intial work of hers. We lived the development of the method together, having better and worse moments, discussing and improving. I want to thank her most of all for having encouraged me when the things seemed to get worse.

I want to show my gratitude to Olivier Saut for having introducing me to the domain of tumor growth modelling. The application of the method in the framework of his computational platform has been a strong boost for my work.

My sincere thanks to Cédric Galusinski and to Gabriella Puppo for having read and reported on the thesis. I really appreciate their advices and I am honored that my work has found their positive judgement.

I give thanks to all the MC2 équipe for the great welcome and the stimulating thirteen months spent working in it. Starting from its director, Thierry Colin, I want to thank Yannick, Jessica, Iraj, and all the others. Among them, a very special thanks to Michel Bergmann for having introducing me to Navier-Stokes simulation and for having made me taste the best Pessac-Leognan I've ever tasted. I will always remeber the wonderful "holydays" in Maratea and in Porquerolles.

Thanks to Mathieu Specklin for a very nice and productive work period together. His help in developing the code was fundamental and his goodbye-party the best party I've been to in Bordeaux.

Another very special thanks goes to Damiano Lombardi. Sharing our works, ideas and passion for athletics was very interesting and enriching.

During the Bordeaux period I found very good friends: Franck, Adrien, Peng, Cédric, Michele, Johanna. Thanks a lot for your special help and all the funny moments together.

I want to give thanks to my office mate for the important and interesting english conversations about almost everything. He kept my english in training, correcting and improving me. It was very important, thanks Farhad.

A special thanks to Florian: his revision of the french (and not only) part of this manuscript has been very important.

It is so hard for me to limit in the word thanks all my gratitude to two very special guys. The reasons are uncountable, scientifically and humanly. I met them in the Sala Alfa. Thanks a lot, Haysam and Federico, for the past and for the future.

This experience has given me a new dear friend, a football team mate and a collegue. His advices have been precious. I want to thank Edoardo.

A very special thanks to Miguel for all the things we did together before the beginning of this work and most of all for having urged me to start this adventure.

Thanks to all the Champisti: Davide, Mara, Roberta, Caterina, Sabino for your friendship and support.

Coming back home, even if just for a weekend, is always a special moment if you have wonderful old friends waiting for you. A huge "thank you" to all my mates: Davide, Stette, Jalla, Umberto, Marcella, the whole Velvet Club, the Buenavista team, Stani, Davide.

For all the wholehearted support they have always given me and for a lot of other things I want to thank my family.

At the end, the most important thanks. No matter where I was or what I was doing, my thoughts were for her, always. She supported me in any case. Staying away from home was hard for both me and her. I want to dedicate all my efforts to express her my infinite gratitude for having waited for me all the times and for all the love she gives me everyday. Thank you, Stefania.

Mobility Scholarships

This work has been conducted in the framework of a co-tutorship between the Politecnico di Torino and l'Université Bordeaux 1. The mobility between this two institution has been possible thanks to the contributions of the following scholarships:

Bando UIF 2009
Contributi per il sostegno alla mobilità de dottorandi
Università Italo-Francese

UNIVERSITÉ FRANCO ITALIENNE | UNIVERSITÀ ITALO FRANCESE

Aide à la mobilité à l'international - Année 2011
Institut Polytechnique de Bordeaux

IPB INSTITUT POLYTECHNIQUE BORDEAUX

Programma Erasmus
2010/2011

Education and Culture
Lifelong Learning Programme
ERASMUS

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Elliptic problems with discontinuous coefficients and sources are often encountered in fluid dynamics, heat transfer, solid mechanics, electrodynamics, material science and biological modelling.
These physical problems have solutions consisting of several components separated by interfaces and for that reason they are often referred as elliptic interface problems. Such interfaces can be still or dynamically moving physical boundaries, material interfaces, phase boundaries, etcetera.
A lot of efforts have been made dealing with these problems, using several different approaches and discretization techniques: finite elements methods on adaptive grids, body-fitted finite difference and finite volume methods and Cartesian grid methods. The present work deals with the latter of these methods by combining it with finite difference schemes, the introduction of new auxiliary unknowns and a dimension-splitting approach.
Many other methods have been designed for solving elliptic interface problems on structured grids. In this work the aim is focused on simplicity of interface treatment, in order to exploit all the Cartesian grid advantages in guaranteeing an easy parallelization.
The high topology complexity of the interface and the need for solving an elliptic interface problem at each time step of a time integration method require that efficiency the parallel computing can give. On the other hand, the literature in elliptic interface problem area lacks in parallel methods which guarantee second order error convergence rate and sharp solutions. This motivated the present study, leading to the development of a parallel second order method for elliptic interface problems which gives accurate sharp solutions across the interface and is easy to be implemented by the use of already existing tools.
The widespread presence of the elliptic interface problems in many scientific domains, we mentioned at the beginning of this section, ensures several different application frameworks. Among them, the present method will get involved in modelling interface phenomena such as free surface dynamics, fluid-structure interaction or electric potential in biological cells. In the present work the application of the method to the elliptic irregular domain problem, close related to the elliptic interface problem, is provided.
Embedding an irregular domain in a regular one and considering boundary conditions as interface jump conditions the present method is exploited in a penalty method spirit to impose homogeneous Neumann conditions on a complex lung

surface, in order to solve pressure and nutrients equations in the framework of a tumor growth model. The lack in second order error convergence rate, up to the irregular lung boundary, of the methods previously employed in this model motivated this application.

1.2 An overview of the work

In this section an overview of the entire work is given. Without details, the main idea of the method and its parallelization are provided and we also touch on tumor growth model, showing details, results and simulations later on.

The aim of this work is to solve the problem, known as elliptic interface problem, described by the following system of partial differential equations

$$\nabla \cdot (k\nabla u) = f \qquad \text{on } \Omega = \Omega_1 \cup \Omega_2 \tag{1.1}$$

$$[\![u]\!] = \alpha \qquad \text{on } \Sigma \tag{1.2}$$

$$[\![k\frac{\partial u}{\partial n}]\!] = \beta \qquad \text{on } \Sigma \tag{1.3}$$

with an order two accuracy on Cartesian grids, using finite differences schemes. $[\![\cdot]\!]$ means $\cdot_1 - \cdot_2$. Ω is the whole computational domain and it is the set union of two sub-domains, Ω_1 and Ω_2, sharing a co-dimension 1 set, Σ, a complex interface. Opportune boundary conditions on $\partial\Omega$, the Ω boundary, complete the system. Figure 1.1 shows an example of this kind of domain.

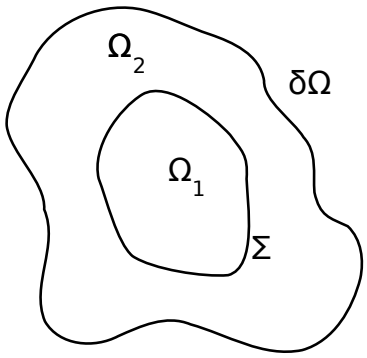


Figure 1.1: Geometry considered: two sub-domains Ω_1 and Ω_2 separated by a complex interface Σ

In this problem k, the diffusion coefficient, f, the source, u, the solution and $k\frac{\partial u}{\partial n}$, the co-normal derivative of the solution could have strong discontinuities across the interface.

Starting from an analysis of the convergence error, in terms of the truncation error, applied to the Laplacian operator in a ghost-point methods spirit, we deduce the needs for order two accuracy:

- a discretization of the Laplacian operator near the interface with a truncation error of order one,
- a discretization of the transmission conditions (1.2) and (1.3) at the interface with a truncation error of order two.

Therefore, we distinguish grid points between regular grid points, far more than a grid step from the interface, and interface grid points, close to the interface less than a grid step (Figure 1.2 gives an example). On the former we discretize the Laplacian operator using standard centred second order finite differences scheme. As far as the latter is concerned, in order to satisfy the order two accuracy requirements, we decide to introduce new unknowns, i.e. the values of the solution at the intersections of the interface with the grid axes (see Figure 1.2).

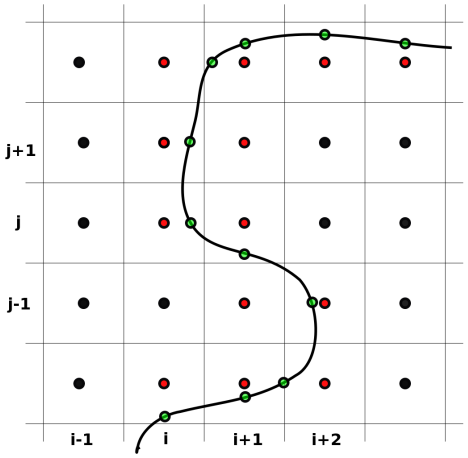
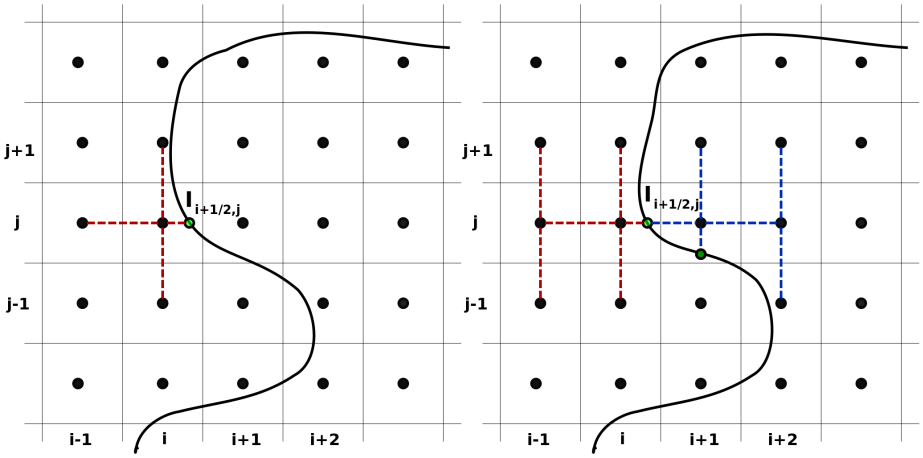


Figure 1.2: Grid points classification: regular grid points are in full black, interface grid points are in red-black and intersection points are in green-black. Straight lines mark cells off.

All these elements allow us to discretize on a Cartesian grid the whole system of equations (1.1 - 1.3). A sketch of the discretization stencils used near the interface for the Laplacian operator and at the interface for the co-normal solution derivative jump is given in Figure 1.3. Solution jump condition is embedded in equations (1.1) and (1.3) discretization as a contribution to the right-hand side.

Clearly, considering the nature of the approximation near the interface, we need information about the interface position and normals. Furthermore, the new variables need to be numbered, in order to assembly the linear system. For the sake of parallelism, the better way to keep this information efficiently is to use a level set function for all the geometrical features of the interface and to store intersections numbering and position in a grid-based parallel data structure.

Two remarks are necessary. Firstly, it is not always possible to discretize the transmission condition (1.3) with a second order truncation error; as the matter of fact doing it needs at least two aligned points on the same side of the interface; when it is not possible a first order truncation error discretization is used. Secondly, the tangential derivative jump condition at the interface can



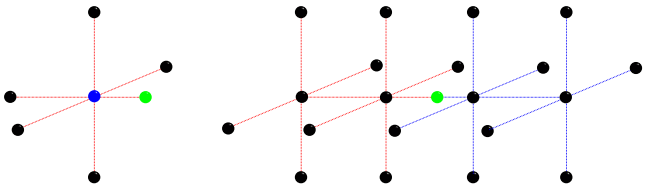
(a) Laplacian operator stencil (b) Equation 1.3 stencil

Figure 1.3: 2D stencils used near the interface.

be deduced from the solution jump condition and used to reduce the number of points involved in the stencil in Figure 1.3b. Some tests have been made and no significant improvements in solution accuracy or in computational performance have been observed.

Preliminary tests, using the scheme we have just touched on here, show instabilities in error convergence rate. For this reason we decide to slightly modify the condition (1.3) stencil, preventing the presence of too close intersections. This gives far smoother error convergence curves.

The extension of the method to the three-dimensional problem is straightforward, thanks to the dimensional splitting approach. An example of the stencils involved in this problem is in Figure 1.4.



(a) Laplacian operator stencil (b) Equation 1.3 stencil

Figure 1.4: 3D stencils used near the interface. Green balls for intersections.

As far as the parallelization of the method is concerned, the use of the level set function and storing out-of-grid points information (the intersection points) as properties of the grid points allow an easy application of the local memory paradigm for parallelism, by using the PETSc library. The domain decomposition is accomplished just paying attention to the intersections numbering. This arrangement is performed trying to reduce the number of message passing com-

munications and keeping the processors load disproportion as low as we can. Finally the parallel code is written using already existing tool, almost without explicitly taking care of the communications.

The numerical validation of the method is carried out providing convergence results for five different two-dimensional test cases and for a preliminary easy three-dimensional test case, using the parallel and the sequential implementations. Error convergence rates show a satisfying second order accuracy even on complex interfaces and the comparisons with other methods in literature prove a competitive absolute error.

The scalability performances of the parallel implementation are tested for both the two-dimensional and the three-dimensional problem, considering the simplest test cases. The results show a good scaling between the computational time and the number of processors, not too far from the perfect parallelism.

Our method is consequently applied to solve the elliptic irregular domain problems appearing in a three-dimensional continuous tumor growth model, specifically the two-species Darcy model.

The continuous model, by means of PDEs, can take into account the space evolution, providing information about the shape and the location of the tumor during its evolution. Among these models the most complex ones are usually not suitable for clinic applications, because of their huge number of free parameters to be determined.

The present application has to be considered in the framework of free parameters identification problem. In this framework inverse problems have to be solved and too many free parameters can make this task computationally impossible. The two-species Darcy model is able to give a reasonable description of the phenomena, surely not considering certain biological mechanisms, but because of its reduced number of free parameters relative to more sophisticated models, it is able to offer an affordable identification problem.

The model describes a three phases saturated flow in a porous isotropic non-uniform medium. P, Q and S are, respectively, the number of proliferating (dividing, responsible for the tumor growth), quiescent and healthy cells per unit volume and the equations for them read:

$$\frac{\partial P}{\partial t} + \nabla \cdot (\vec{v}P) = (2\gamma - 1)P + \gamma Q \tag{1.4}$$

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\vec{v}Q) = (\gamma - 1)P - \gamma Q \tag{1.5}$$

$$\frac{\partial S}{\partial t} + \nabla \cdot (\vec{v}S) = 0 \tag{1.6}$$

where the velocity $\vec{v}$ accounts for the tissue deformation and γ, a scalar function of the nutrients, determines if the tumor cells proliferate or die (becoming quiescent). Passive motion is assumed. The saturated flow hypothesis is used, $P + Q + S = 1$, which implies

$$\nabla \cdot \vec{v} = \gamma P \tag{1.7}$$

and the mechanical clusure of the system is given by a Darcy law for the velocity

$$\vec{v} = -k(P, Q)\nabla \Pi \tag{1.8}$$

where Π is a scalar function playing the role of a pressure (or potential) and k is the permeability field given by

$$k = k_1 + (k_2 - k_1)(P + Q) \tag{1.9}$$

being k_1 the constant permeability of the healthy tissue and k_2 the constant permeability of the tumor tissue.

The nutrients (specifically, the oxygen) are governed by a diffusion-reaction equation:

$$\partial_t C - \nabla \cdot (D(P,Q)\nabla C) = 0.1(C_{max} - C)S - \alpha PC - 0.01\alpha QC \tag{1.10}$$

where C is the density of nutrients, $C_{max} = 1$ and α is the nutrients consumption rate for the proliferating cells.

$$D = D_{max} - K(P + Q) \tag{1.11}$$

is the diffusivity expressed by a phenomenological law reflecting the different diffusion of the oxygen in healthy and tumor tissues.

The γ function expresses the transition between the proliferating and quiescent state and it is the regularization of the unit step

$$\gamma = \frac{1 + \tanh(R(C - C_{hyp}))}{2} \tag{1.12}$$

where R is a coefficient and C_{hyp} is the hypoxia threshold.

The domain, Ω, is generally a complex domain with an irregular boundary. No mass can leave this domain and for this reason homogeneous Neumann boundary conditions are imposed for both the oxygen and the pressure equations. But the Neumann problem for the pressure has to be well posed, then a modification of the velocity divergence is needed: it has to be a zero average scalar quantity and so

$$\nabla \cdot \vec{v} = \gamma(C) - \frac{\int_\Omega \gamma P \, d\Omega}{\int_\Omega 1 - P - Q \, d\Omega}(1 - P - Q). \tag{1.13}$$

This correction means a compression of the healthy tissue caused by the tumor growth and consequently the equation (1.6) can no longer hold.

Our method is introduced to solve the equation for the pressure (1.8) and for the nutrients (1.10) in an irregular domain. The original domain Ω is embedded in a larger regular one Ω'. The original boundary conditions are considered as transmission conditions at the interface separating Ω and its relative component in Ω', introducing new permeability and diffusivity

$$k' = \begin{cases} k, & \text{in } \Omega \\ k_{out}, & \text{in } \Omega' \setminus \Omega \end{cases} \tag{1.14}$$

$$D' = \begin{cases} D, & \text{in } \Omega \\ D_{out}, & \text{in } \Omega' \setminus \Omega \end{cases} \tag{1.15}$$

In the spirit of a penalty method we impose approximated homogeneous Neumann boundary conditions for pressure and oxygen at the boundary of Ω, choosing very small values for k_{out} and D_{out}.

We perform two simulations of the model with different geometries and initial conditions. The aim is to show the importance of three-dimensional modelling in tumor growth and the role played by the boundary of Ω in the evolution of a tumor nodule shape. The interfaces (boundaries of Ω) chosen are a sphere and a lung. The latter is obtained by segmentation of medical images (CT-scans). The results provide, on a quality level, a good behaviour of the method in imposing the homogeneous Neumann boundary conditions on the border of Ω and a reasonable evolution of the nodule shape and composition.

Error convergence studies with more complex three-dimensional interfaces as well as quantitative results and comparisons with realistic cases in tumor growth simulation are needed to corroborate what is shown in the present work. Managing more than one interface, then more than two different sub-domains is an important improvement not only for the method itself, but for the tumor growth model too. This can introduce the possibility of considering a structured, and therefore more realistic, original domain. These are the main future perspectives of this work.

1.3 Structure of the work

In CHAPTER 4 the parallel second order method for elliptic interface problems is introduced.
In *section 4.1* an overview about the existing and the present method is given.
The successive *section 4.2* provides the genesis of the main idea, analysing the convergence error for the one-dimensional problem.
In *section 4.3* the detailed description of the method is illustrated for the two-dimensional problem.
Afterwards, in *section 4.4*, the dimensional splitting is used to give some details about the method for the three-dimensional problem.
The parallel implementation is introduced in *section 4.5* with highlights on the model adopted, on the tools used and on the arrangement of the linear system.
The *section 4.6* provides the numerical validation of the method, showing the error convergence rate results for a good range of two-dimensional test cases, using both sequential and parallel implementations. A simple three-dimensional test case and the scalability performances of the parallel implementation are also discussed in this section.
The chapter ends with the conclusions about the method in section 4.7.
In CHAPTER 5 the method is applied to solve the elliptic irregular domain problem in the framework of a tumor growth model.
In *section 5.1* a brief introduction to tumor growth modelling is given and the chosen model is illustrated in section *5.2*.
Afterwards, in *section 5.3* the whole numerical application framework is introduced. We show how our method can be used to solve the elliptic irregular domain problem and some notes about medical imagery and segmentation tools are provided in order to show how the geometries involved have been obtained.
Finally, in *section 5.4* the results of the simulations are shown and discussed.
Conclusions about the application conclude the chapter, in *section 5.5*.
Perspectives and future intentions about the method itself, the tumor growth model and the alternative applications of the method are discussed in CHAPTER 6.

Chapter 2

Introduzione (italiano)

2.1 Motivazione

I probelmi ellittici con sorgenti e coefficienti discontinui sono comuni nella fluidodinamica, nella trasmissione del calore, in meccanica dei solidi, in elettrodinamica, nelle scienze dei materiali e nella modellistica biologica.
Le soluzioni di questi problemi fisici consistono in diverse componenti separate da interfacce e per questo motivi sono spesso conosciuti come problemi ellittici con interfacce. Tali interfacce possono essere contorni fisici stazionari o mobili, interfacce materiali, contorni di fase, eccetera.
Molti sforzi sono stati compiuti trattando questi problemi, facendo uso di molti e differenti approcci e tecniche di discretizzazione: metodi agli elementi finiti su griglie adattative, metodi ai volumi finiti e alle differenze finite su griglie adattate e metodi su griglia cartesiana. Questo lavoro tratta l'ultimo di questi metodi combinandolo con gli schemi alle differenze finite, con l'introduzione di nuove incognite ausiliarie e con un approccio dimensione-per-dimensione.
Molti altri metodi sono stati sviluppati per risolvere i problemi ellitici con interfacce su griglie strutturate. In questo lavoro vogliamo porre l'accento sulla semplicità di trattamento dell'interfaccia, al fine di sfruttare tutti i vantaggi di una griglia cartesiana garantendo una facile parallellizzazione.
L'elevata complessità topologica dell'interfaccia e la necessità di risolvere un problema ellittico con interfaccia ad ogni passo temporale di un metodo di integrazione in tempo richiedono un livello di efficienze che il calcolo parallelo può fornire. D'altro canto, la letteratura relativa al problema ellitico con interfaccia è priva di metodi paralleli che garantiscano un tasso di convergenza dell'errore del secondo ordine e soluzioni nette all'interfaccia. Ciò ha motivato il presente studio, portandoci allo sviluppo di un metodo al secondo ordine parallelo per problemi ellittici con interfacce capace di fornire soluzioni nette e discontinue attraverso l'interfaccia e facile da implementare grazie all'uso di strumenti esistenti.
La diffusa presenza dei problemi ellittici con interfacce in molti campi scientifici, menzionata all'inizio di questa sezione, assicura numerosi e differenti contesti applicativi. Tra questi, il metodo attuale verrà impiegato nella modellizzazione di fenomeni d'interfaccia quali la dinamica di un'interfaccia libera, l'interazione fluido-struttura o lo studio del potenziale elettrico in cellule biologiche. In questo lavoro l'applicazione del metodo al problema ellittico in dominio irregolare, strettamente connesso al problema ellittico con interfaccia, è fornita.
Inserendo un dominio irregolare in uno regolare e considerando le condizioni

al bordo come condizioni di salto, il presente metodo è sfruttato nello spirito di un metodo di penalizzazione per imporre condizioni di Neumann omogenee sulla superficie di un polmone complesso, al fine di risolvere le equazioni per la pressione e per i nutrienti nel contesto di un modello di crescita tumorale. La mancanza di un tasso di convergenza al secondo ordine dell'errore, in regioni prossime al contorno del polmone, nei metodi precedentemente impiegati in questo modello ha motivato questa applicazione.

2.2 Una visione d'insieme del lavoro

In questa sezione viene fornita una panoramica dell'intero lavoro. Senza dettagli, proponiamo l'idea principale del metodo e la sua parallelizzazion e accenniamo al modello di crescita tumorale, mostrando dettagli, risultati e simulazioni piú avanti.

Lo scopo di questo lavoro è risolvere il problema, noto come problema ellittico con interfacca, descritto dal seguente sistema di equazioni alle derivate parziali

$$\nabla \cdot (k \nabla u) = f \qquad \text{on } \Omega = \Omega_1 \cup \Omega_2 \tag{2.1}$$

$$[\![u]\!] = \alpha \qquad \text{on } \Sigma \tag{2.2}$$

$$[\![k \frac{\partial u}{\partial n}]\!] = \beta \qquad \text{on } \Sigma \tag{2.3}$$

al secondo ordine di accuratezza su griglie cartesiane, utilizzando schemi alle differenze finite. $[\![\cdot]\!]$ sta per $\cdot_1 - \cdot_2$. Ω è l'intero dominio computazionale ed è l'unione di due sottodomini, Ω_1 e Ω_2, che condividono un insieme di codimensione 1, Σ, un interfaccia complessa. Opportune condizioni al bordo su $\partial\Omega$, il bordo di Ω, completano il sistema. Figura 2.1 mostra un esempio di questo tipo di dominio.

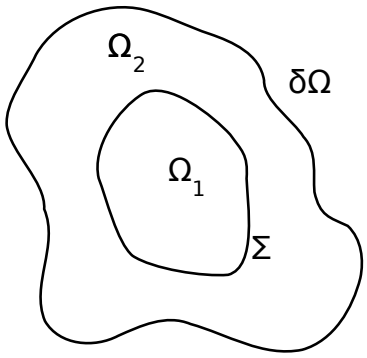


Figure 2.1: Geometria considerata: due sottodomini Ω_1 e Ω_2 separati da un'interfaccia complessa Σ

In questo problema, k, il coefficiente di diffusione, f, la sorgente, u, la soluzione e $k\frac{\partial u}{\partial n}$, la derivata conormale della soluzione potrebbero avere forti discontinuità attraverso l'interfaccia.

Partendo dall'analisi dell'errore di convergenza, in termini dell'errore di troncamento, applicata all'operatore laplaciano nello spirito dei metodi ghost-point, deduciamo quali richieste dobbiamo soddisfare per ottenere un'accuratezza al secondo ordine:

- una discretizzazione nei pressi dell'interfaccia dell'operatore di Laplace con un errore di troncamento di ordine uno.
- una discretizzazione delle condizioni di trasmissione (2.2) e (2.3) all'interfaccia con un errore di troncamento di ordine due.

Perciò, distinguiamo i punti della griglia in punti regolari, lontani piú di un passo griglia dall'interfaccia, e in punti di interfaccia, a meno di un passo griglia dall'interfaccia (Figura 2.2 dà un esempio). Sui primi discretiziamo l'operatore di Laplace usando lo schema alle differenze finite centrato standard al secondo ordine. Per quanto riguarda i secondi, al fine di soddisfare le richieste relative all'accuratezza di ordine due, decidiamo di introdurre nuove incognite, ovvero i valori della soluzione nelle intersezioni dell'interfaccia con gli assi della griglia (see Figure 2.2).

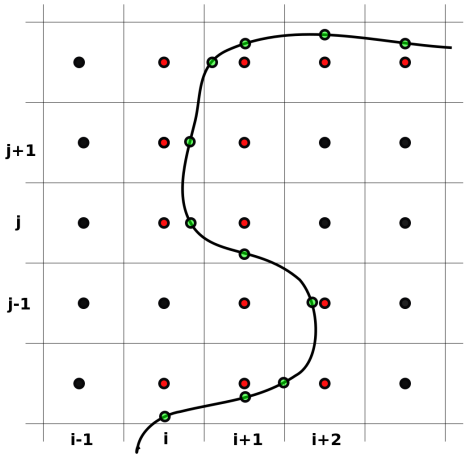
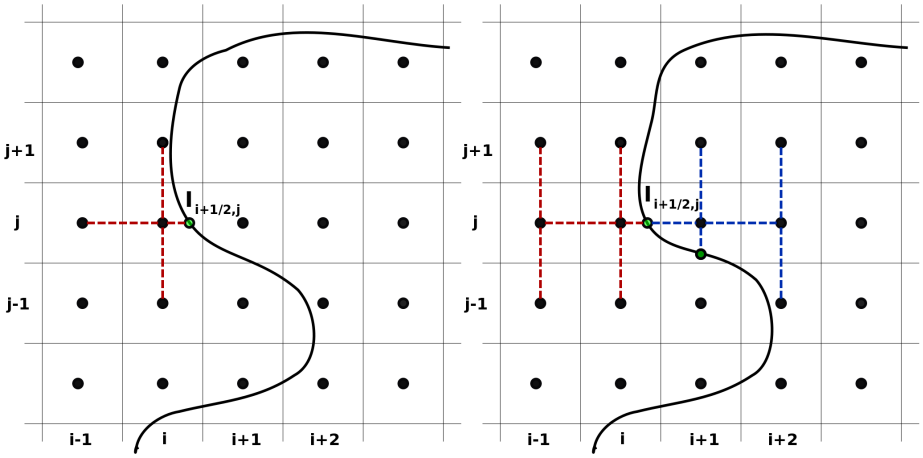


Figure 2.2: Classificazione dei punti griglia: i punti griglia regolari in nero pieno, punti griglia d'interfaccia sono in rosso-nero e i punti d'intersezione in verde-nero. Linee continue contornano le celle.

Gli elementi sin qui introdotti ci permettono di discretizzare l'intero sistema di equazioni (2.1 - 2.3) su una griglia cartesiana. Una rappresentazione grafica degli stencils di discretizzazione utlizzati nei punti vicini all'interfaccia per l'operatore di Laplace e sull'interfaccia per il salto della derivata conormale della soluzione è proposta in Figura 2.3. La condizione di salto per la soluzione è inserita nella discretizzazione delle equazioni (2.1) and (2.3) come contributo al termine noto.

È evidente che, considerando la natura dell'approssimazione, abbiamo bisogno di informazione sulla posizione dell'interfaccia e sulle normali all'interfaccia stessa. Inoltre, le nuove variabili devono essere numerate per poter consistentemente assemblare il sistema lineare. Per una buona implementazione parallela, il modo migliore di conservare efficientemente tali informazione è di utilizzare



(a) Stencil per l'operatore di Laplace (b) Stencil per l'equazione 2.3

Figure 2.3: Stencils 2D impiegati nei pressi dell'interfaccia.

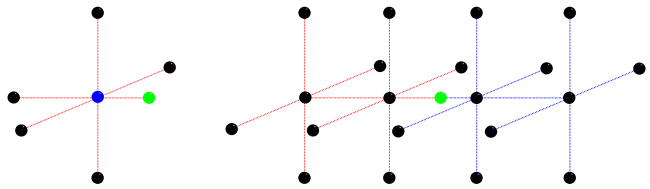
una funzione level set per tutte le caratteristiche geometriche dell'interfaccia e di immagazzinare la numerazione e la posizione delle intersazioni in una struttura parallela basata sulla griglia.

A questo punto sono necessarie due osservazioni. In primo luogo, non è sempre possibile discretizzare la condizione di trasmissione (2.3) con un errore di troncamento al secondo ordine; infatti per far ciò abbiamo bisogno di almeno due punti allineati sullo stesso lato dell'interfaccia; quando questo non è possibile una discretizzazione con errore di troncamento al primo ordine viene utilizzata. In secondo luogo, la condizione di salto all'interfaccia per la derivata tangenziale della soluzione può essere dedotta dalla condizione di salto della soluzione e utilizzata per ridurre il numero di punti coinvolti nello stencil in Figura 2.3b. Alcuni tests hanno dimostrato che nessun miglioramento significativo nell'accuratezza della soluzione o nella prestazione computazionale può essere apprezzato.

Tests preliminari, utilizzando lo schema sin qui accennato, mostrano instabilità nel tasso di convergenza dell'errore. Per questo motivo decidiamo di modificare leggermente la discretizzazione della condizione (2.3), impedendo la presenza di intersezioni troppo vicine nello stesso stencil. Ciò fornisce curve di convergenza dell'errore molto piú lisce.

L'estensione del metodo al problema tridimensionale è semplice considerando ogni direzione in maniera indipendente. Un esempio degli stencils coinvolti in tale problema è dato in Figura 2.4.

Per quanto riguarda la parallelizzazione del metodo, l'uso della funzione level set e l'immagazzinare l'informazione dei punti fuori griglia (i punti d'intersezione) come proprietà dei punti della griglia permettono una facile applicazione del paradigma di programmazione parallela a memoria locale, attraverso l'uso della libreria PETSc. La decomposizione di dominio è ottenuta semplicemente facendo attenzione alla numerazione delle intersezioni. Tale numerazione è compiuta cercando di ridurre il numero di comunicazioni message passing and di tenere il divario tra i carichi dei processori il piú basso possibile. Infine il codice parallelo



(a) Stencil per l'operatore di Laplace (b) Stencil per l'equazione 2.3

Figure 2.4: Stencils 3D impiegati nei pressi dell'interfaccia. Le sfere verdi rappresentano le intersazioni.

è scritto utilizzando strumenti già esistenti, quasi senza occuparsi esplicitamente delle comunicazioni.

La validazione numerica del metodo è realizzata fornendo risultati di convergenza per cinque diversi casi bidimensionali a per un semplice e preliminare caso tridimensionale, utilizzando le implementazioni sequanziale e parallela. I tassi di convergenza dell'errore mostrano una soddisfacente accuratezza al secondo ordine anche su interfacce complesse e i confronti con altri metodi in letteratura provano un errore assoluto competitivo.

Le prestazioni di scalabilità dell'implementazione parallela sono testate sia per il problema bidimensionale che per quello tridimensionale, considerando i casi piú semplici. I risultati mostrano un buon comportamento del tempo di calcolo in funzione del numero di processori, non troppo lontano dal parallelismo perfetto.

Il nostro metodo è conseguentemente applicato per risolvere i problemi ellittici in dominio irregolare che compaiono nel modello continuo e tridimensionale di crescita tumorale, specificatamente il modello a due specie di Darcy.

Il modello continuo, per mezzo di EDP, è in grado di considerare l'evoluzione spaziale, fornendo informazioni sulla forma e la localizzazione del tumore durante il suo sviluppo. Tra questi modelli i piú complessi non sono solitamente adatti ad applicazioni cliniche, a causa dell'elevato numero di parametri liberi da determinare.

La presente applicazione deve essere considerata nel contesto del problema di identificazione dei parametri liberi di un modello. In questo contesto è necessario risolvere dei problemi inversi e troppi parametri liberi possono rendere questo obiettivo computazionalmente impossibile. Il modello a due specie di Darcy può dare una ragionevole descrizione del fenomeno, pur non considerando alcuni meccanismi biologici, ma a causa del suo ridotto numero di parametri liberi rispetto a modelli piú sofisticati, è in grado di offrire un accessibile problema di identificazione.

Il modello descrive il flusso saturo di tre fasi in un mezzo poroso non uniforme e isotropo.

P, Q e S sono, rispettivamente, il numero di cellule proliferanti (capaci di dividersi e responsabili della crescita tumorale), quiescenti e sane per unità di

volume e le equazioni che le governano sono:

$$\frac{\partial P}{\partial t} + \nabla \cdot (\vec{v}P) = (2\gamma - 1)P + \gamma Q \tag{2.4}$$

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\vec{v}Q) = (\gamma - 1)P - \gamma Q \tag{2.5}$$

$$\frac{\partial S}{\partial t} + \nabla \cdot (\vec{v}S) = 0 \tag{2.6}$$

dove la velocità $\vec{v}$ rende conto della deformazione tissutale e γ, una funzione scalare dei nutrienti, determina se le cellule tumorali proliferano o muoiono (diventando quiescenti). Si assume moto passivo. L'ipotesi di flusso saturo è introdotta, $P + Q + S = 1$, la quale implica

$$\nabla \cdot \vec{v} = \gamma P \tag{2.7}$$

e la chiusura meccanica del sistema è data da una legge di Darcy per la velocità

$$\vec{v} = -k(P, Q)\nabla \Pi \tag{2.8}$$

dove Π è una funzione scalare che gioca il ruolo di pressione (o potenziale) e k é il campo di permeabilità dato da

$$k = k_1 + (k_2 - k_1)(P + Q) \tag{2.9}$$

essendo k_1 la permeabilità costante del tessuto sano e k_2 la permeabilità costante del tessuto tumorale.

I nutrienti (specificatamente, l'ossigeno) sono governati da un'equazione di reazione-diffusione:

$$\partial_t C - \nabla \cdot (D(P, Q)\nabla C) = 0.1(C_{max} - C)S - \alpha PC - 0.01\alpha QC \tag{2.10}$$

dove C è la densità di nutrienti, $C_{max} = 1$ α è il tasso di consumo di nutrienti ad opera delle cellule proliferanti.

$$D = D_{max} - K(P + Q) \tag{2.11}$$

è la diffusività espressa da una legge fenomenologica che rispecchia le differenti diffusioni dell'ossigeno nei tessuti sano e tumorale.

La funzione γ regola la transizione tra gli stati proliferante e quiescente ed è la regolarizzazione del gradino unitario

$$\gamma = \frac{1 + \tanh(R(C - C_{hyp}))}{2} \tag{2.12}$$

dove R è un coefficiente e C_{hyp} è la soglia di ipossía.

Il dominio, Ω, è generalmente un dominio complesso con un contorno irregolare. La massa interna al dominio non può uscire dal dominio stesso, quindi condizioni al contorno di Neumann omogenee sono imposte sia per l'ossigeno che per la pressione. Ma il problema di Neumann per la pressione deve essere ben posto, quindi è necessario modificare la divergenza della velocità: deve essere una quantità scalare a media nulla nel dominio e dunque

$$\nabla \cdot \vec{v} = \gamma(C) - \frac{\int_\Omega \gamma P \, d\Omega}{\int_\Omega 1 - P - Q \, d\Omega}(1 - P - Q). \tag{2.13}$$

Tale correzione implica una compressione del tessuto sano causata dalla crescita del tumore e conseguentemente l'equazione (2.6) non è piú valida.

Il nostro metodo è introdotto per risolvere le equazioni per la pressione (2.8) e per i nutrienti (2.10) in un dominio irregolare. Il dominio originale Ω viene quindi inserito in uno piú grande e regolare Ω'. Le condizioni al contorno originali sono considerate come condizioni di trasmissione all'interfaccia che separa Ω e il suo complemento relativo in Ω', introducendo nuove permeabilità e diffusività.

$$k' = \begin{cases} k, & \text{in } \Omega \\ k_{out}, & \text{in } \Omega' \setminus \Omega \end{cases} \tag{2.14}$$

$$D' = \begin{cases} D, & \text{in } \Omega \\ D_{out}, & \text{in } \Omega' \setminus \Omega \end{cases} \tag{2.15}$$

Utilizzando un metodo di penalizzazione, imponiamo un'approssimazione delle condizioni al contorno di Neumann omogenee per la pressione e per l'ossigne al bordo di Ω, scegliendo valori per k_{out} e D_{out} molto piccoli.

Realizziamo due simulazioni del modello con differenti geometrie e condizioni iniziali. Lo scopo é di mostrare l'importanza della modellizzazione tridimensionale della crescita tumorale e il ruolo giocato dal contorno di Ω nell'evoluzione della forma del nodulo tumorale. Le interfacce (contorni di Ω) scelte sono una sfera e un polmone. Quest'ultimo è ottenuto per segmentazione di immagini medicali (scansioni di CT). I risultati forniscono, ad un livello qualitativo, un buon comportamento nell'imporre le condizioni al contorno di Neumann omogenee da parte del metodo e una ragionevole evoluzione della forma e della composizione del nodulo.

Studi di convergenza dell'errore con interfacce tridimensionali piú complesse cosí come risultati quantitativi e confronti con casi realistici per quanto riguarda la crescita tumorale sono necessari per corroborare ciò che è mostato nel presente lavoro. These are the main future perspectives of this work. Gestire la presenza di piú interfacce, quindi piú di due sottodomini è un miglioramento importante non solo per il metodo in sé ma anche per il modello di crescita tumorale. Ciò infatti introdurrebbe la possibilità di considerare un dominio originale strutturato e quindi piú realistico. Queste sono le prospettive future principali di questo lavoro.

2.3 Struttura della tesi

Nel CAPITOLO 4 il metodo parallelo al secondo ordine per problemi ellittici con interfaccia è introdotto.
Nella *sezione 4.1* una panoramica dei metodi esistenti e del presente metodo è data.
La successiva *sezione 4.2* fornisce la genesi dell'idea principale, analizzando l'errore di convergenza per il problema unidimensionale.
Nella *sezione 4.3* la descrizione dettagliata del metodo viene illustrata per il problema bidimensionale.
Successivamente, nella *sezione 4.4*, lo splitting dimensionale è usato per fornire qualche dettaglio circa il metodo per i problemi tridimensionali.

L'implementazione parallela è introdotta nella *section 4.5* sottolineando il modello introdotto, gli strumenti di programmazione impiegati e l'organizzazione del sistema lineare.
La *seezione 4.6* fornisce la validazione numerica del metodo, mostrando i risultati sui tassi di convergenza dell'errore per un buon intervallo di casi bidimensonali, utilizzando sia l'implementazione parallela che quella sequanziale. Un semplice caso tridimensionale e le prestazioni di scalabilità dell implementazione paralle sono pure discussi in questa sezione.
Il capitolo termina con le conclusione circa il metodo in sezione 4.7.
Nel CAPITOLO 5 il metodo è applicato per risolvere il problema ellittico in dominio irregolare nel contesto di un modello di crescita tumorale
Nella sezione *section 5.1* una breve introduzione alla modellizzazione di crescita tumorale è data e il modello scelto è illustrato nella sezione *5.2*.
Successivamente, nella *sezione 5.3* l'intero contesto numerico dell'applicazione è introdotto. Mostriamo come il nostro metodo può essere usato per risolvere il problema ellittico in dominio irregolare e qualche cenno alla dignostica per immagini e agli strumenti di segmentazione è fornito per mostrare come le geometrie coinvolte sono state ottenute.
In fine, nella *sezione 5.4* i risultati delle simulazioni sono mostrati e discussi.
Le conclusioni circa l'applicazione concludono il capitolo in *sezione 5.5*.
Le propsettive e le intenzioni future sul metodo, sul modello di crescita tumorale e sulle applicazioni alternative del metodo sono discusse nel CAPITOLO 6.

Chapter 3

Introduction (français)

3.1 Motivation

Les problèmes elliptiques avec coefficients et sources discontinues sont souvent rencontrés dans la mécanique des fluides, la transmission de la chaleur, la mécanique des solides, l'électrodynamique, la science des matériels et la modélisation biologique.
Les solutions de ces problèmes physiques sont faites de plusieurs composantes séparées par des interfaces et pour cette raison ils sont souvent appelés problèmes elliptiques avec interfaces. Telles interfaces peuvent être contours physiques stationnaires ou mobiles, interfaces matérielles, contours de phase, et cetera.
Beaucoup d'efforts ont été faits pour régler ces problèmes en utilisant plusieurs approches et techniques de discrétisation : méthodes à éléments finis sur maillages adaptatifs, méthodes à différences finies et volumes finis sur maillages adaptés au corps et méthodes sur maillages cartésiens.

Ce travail concerne ces derniers en les combinant avec des schémas aux différences finies, l'introduction de nouvelles inconnues et une approche dimension par dimension.

Beaucoup d'autre méthodes ont été conçues pour rèsoudre des problémes elliptiques avec interfaces sur maillages structurés. Dans ce travail le but est la simplicité du traitement de l'interface, pour exploiter tous les avantages des maillages cartésiens pour la parallelisation.

La complexité élevée de la topologie de l'interface et le besoin de résoudre un problème elliptique à chaque pas de temps d'une méthode d'intégration en temps demandent une efficacitÃ© que le calcul parallèle peut apporter. D'autre part, la littérature concernant le problème elliptique avec interface manque de méthodes parallèles garantissant un taux de convergence de l'erreur d'ordre deux et des solutions nettes sur l'interface. Tout cela a motivé la présente étude, conduisant au développement d'une méthode au deuxième ordre parallèle pour problèmes elliptiques avec interfaces donnant des solutions nettes et précises à travers l'interface et qui est simple à être implémenté grâce á des instruments existants.

La présence étendue des problèmes elliptiques avec interfaces dans plusieurs domaines scientifiques , dont on a parlÃ© au début de cette section, assure un grand nombre de cadres d'application. Parmi ces cadres, la méthode actuelle sera utilisée dans la modélisation de phénomènes avec interfaces comme la dynamique d'une interface libre, l'interaction fluide-structure ou l'évolution du potentiel électrique dans les cellules biologiques. Dans ce travail on fournit

l'application de la méthode à le problème elliptique dans un domaine irrégulier, strictement lié au problème elliptique avec interface.

En introduisant un domaine irrégulier dans un domaine régulier et en considérant les conditions aux bords comme des conditions sur les sauts à l'interface la méthode actuelle est exploitée dans l'esprit d'une méthode de pénalisation pour imposer conditions de Neumann homogènes sur la surface d'un poumon complexe, afin de résoudre les équations pour la pression et les nutriments dans le cadre d'un modèle de croissance tumorale. L'absence d'un taux de convergence de l'erreur au deuxième ordre jusqu'au bord complexe du poumon dans les méthodes utilisées précédemment a motivé cette application.

3.2 Une vue d'enseble du travail

Dans cette section on donne une vue d'ensemble du travail. L'idée principale de la méthode et de sa parallélisation est donnée et on présente le modèle de croissance tumorale, en montrant les détails, les résultats et les simulation plus loin dans ce rapport.

Le but de ce travail est de résoudre le problème, connu comme le problème elliptique avec interface, défini par le suivant système d'équations aux dérivées partielles.

$$\begin{aligned} \nabla \cdot (k \nabla u) &= f \qquad \text{on } \Omega = \Omega_1 \cup \Omega_2 && (3.1)\\ [\![u]\!] &= \alpha \qquad \text{on } \Sigma && (3.2)\\ [\![k \frac{\partial u}{\partial n}]\!] &= \beta \qquad \text{on } \Sigma && (3.3) \end{aligned}$$

avec une précision d'ordre deux sur maillage cartésien et en utilisant des schémas de différences finies. $[\![\cdot]\!]$ veut dire $\cdot_1 - \cdot_2$. Ω est le domaine entier de calcul et c'est l'union de deux sous-domaines Ω_1 and Ω_2, partageant un ensemble de co-dimension 1, Σ, une interface complexe.

D'opportunes conditions aux bords sur $\partial\Omega$, le bord de Ω, complètent le système. La Figure 3.1 montre un exemple de ce type de domaine.

Dans ce problème k, le coefficient de diffusion, f, la source, u, la solution et $k\frac{\partial u}{\partial n}$, la dérivée conormale de la solution, pourraient avoir de fortes discontinuités à travers l'interface.

En commençant par l'analyse de l'erreur de convergence, en fonction de l'erreur de troncature, appliquée à l'opérateur de Laplace dans l'esprit d'une méthode ghost-point, on déduit que pour avoir une précision d'ordre deux il faut:

- une discrétisation de l'opérateur de Laplace sur les points proches de l'interface avec un erreur de troncature d'ordre 1
- une discrétisation des conditions de transmission (3.2) et (3.3) sur l'interface avec un erreur de troncature d'ordre 2.

On distingue donc les points du maillage entre points réguliers, à plus d'un pas d'espace de l'interface, et points d'interface, à moins d'un pas d'espace de l'interface (Figure 3.2 donne un exemple). Sur les premiers on discrétise l'opérateur de Laplace en utilisant le schéma standard aux différences finies

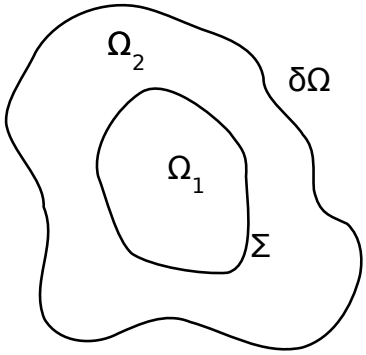


Figure 3.1: Géométrie considérée : deux sous-domaines Ω_1 et Ω_2 séparés par une interface complexe Σ

d'ordre deux. Pour les second, afin de satisfaire les demandes relatives au deuxième ordre, on introduit des nouvelles inconnues, c'est à dire les valeurs de la solution aux intersections de l'interface avec les axes du maillage (Figure 3.2).

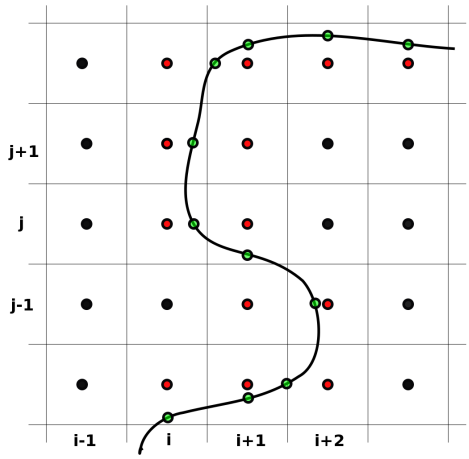
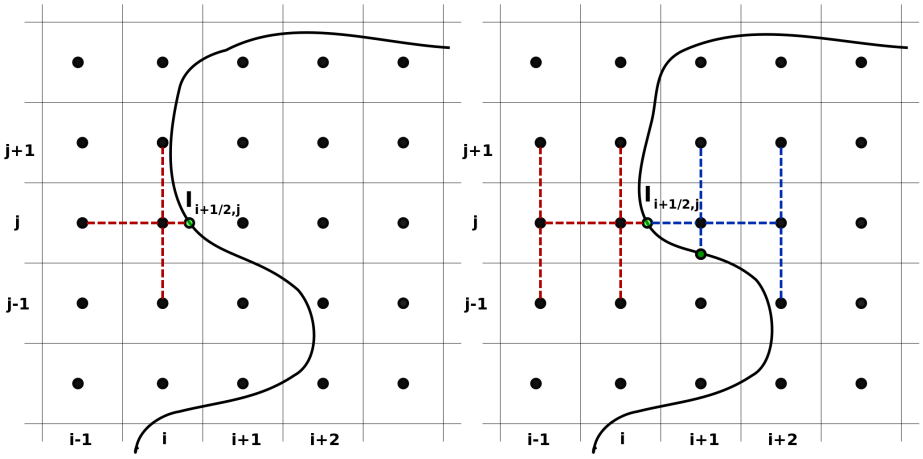


Figure 3.2: Classification des points du maillage : les points réguliers sont en noir, les points d'interface sont en rouge et noir et les intersections sont en vert. Les lignes continues entourent les cellules.

Tous ces éléments nous permettent de discrétiser sur un maillage cartésien l'entier système d'équations (3.1 - 3.3). Un dessin schématique des stencils de discrétisation utilisés à coté de l'interface pour l'opérateur de Laplace et sur l'interface pour le saut de la dérivée conormale de la solution est montré en Figure 3.3. La condition de saut de la solution est prise en compte dans la discrétisation des équations (3.1) et (3.3) comme une contribution dans le second membre.



(a) Stencil pour l'opérateur de Laplace. (b) Stencil pour l'équation 3.3

Figure 3.3: Stencils 2D utilisés autour de l'interface.

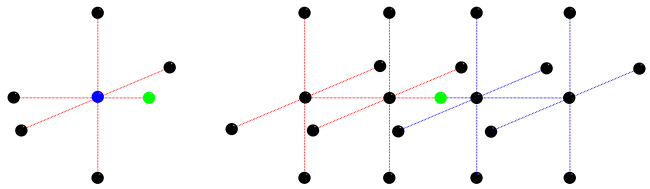
Évidement, en considérant la nature de l'approximation autour de l'interface, il nous faut de l'information sur la position de l'interface et sur ses normales. En plus, les nouvelles inconnues ont besoin d'être énumérées, afin d'assembler le système linéaire. Pour une bonne parallélisation, la meilleure manière de garder cette information de façon efficace est d'utiliser la fonction level set pour toutes les caractéristiques géométriques de l'interface et de stocker l'énumération et la position des intersections dans une structure de données parallèle basée sur le maillage.

Deux remarques sont nécessaires. En premier lieu, il n'est pas toujours possible de discrétiser la condition de transmission (3.3) avec une erreur de troncature d'ordre deux ; en effet pour cela il faut au moins deux points sur la même ligne du même coté de l'interface ; lorsque cela est impossible on utilise une discrétisation avec une erreur de troncature d'ordre un. En second lieu, la condition sur le saut sur l'interface de la dérivée tangentielle peut être déduite de la condition sur le saut de la solution et utilisée pour réduire le nombre des points impliqués dans le stencil en Figure 3.3b. Différents tests ont été réalisés sans pour autant observer d'améliorations significatives que ce soit sur la précision de la solution ou sur le temps de calcul.

Les tests préliminaires, réalisé avec le schéma que l'on viens de présenter dans cette section, montrent des instabilités dans le taux de convergence de l'erreur. Pour cette raison on décide de modifier légèrement le stencil de la condition (3.3), afin de éviter la présence de intersections trop proche l'une de l'autre. Cette modification donne des courbes de convergence de l'erreur beaucoup plus lisses.

La généralisation de la méthode au problème tridimensionnel est simple, grâce à l'approche dimension-par-dimension. Un exemple des stencils impliqués dans ce problème est montré en Figure 3.4.

En ce qui concerne la parallélisation de la méthode, l'utilisation de la fonction level set et le stockage de l'information sur les points hors du maillage (les intersections) comme une caractéristique des points du maillage permettent



(a) Stencil pour l'opérateur de Laplace.

(b) Stencil pour l'équation 3.3.

Figure 3.4: Stencils 3D utilisés autour de l'interface. Sphères vertes pour les intersections.

une application simple du paradigme à mémoire locale pour le parallélisme, en employant la bibliothèque PETSc. La décomposition de domaine est obtenue apportant un soin particulier juste à l'énumération des intersections. Cette organisation est réalisée en cherchant á réduire le nombre de communications message passing et de garder le déséquilibre parmi les travaux de chaque processeurs le plus faible possible. Finalement, le code parallèle est écrit en employant des instruments de codage parallèle déjà existants, presque sans s'occuper explicitement des communications.

La validation numérique de la méthode est réalisée en produisant les résultats de convergence pour cinq différents cas-test bidimensionnels et pour un simple cas-test tridimensionnel préliminaire, en utilisant les implémentations séquentielle et parallèle du code. Les taux de convergence de l'erreur montrent une précision satisfaisante d'ordre deux même pour des interfaces complexes et les comparaisons avec les autre méthodes présentes dans la littérature montrent une erreur absolue compétitive.

Les performances de scalabilité de l'implémentation parallèle sont testées que ce soit pour le problème bidimensionnel ou pour le problème tridimensionnel, en considérant le cas-test le plus simple. Les résultats montrent un bon comportement du temps de calcul en fonction du nombre des processeurs, peu éloigné du parallélisme parfait.

Notre méthode est ensuite appliquée pour résoudre les problèmes elliptiques dans un domaine irrégulier apparissant dans un modèle continu tridimensionnel de croissance tumorale, en particulier le modèle à deux espèces de Darcy. Le modèle continu, au moyen d'EDP, peut traiter l'évolution spatiale d'une tumeur pendant son développement en fournissant des informations sur sa forme et sa position. Parmi ces modèles les plus complexes ne sont normalement pas adaptes pour des applications cliniques, à cause du nombre élevé de paramètres libres à déterminer.

L'application actuelle doit être considérée dans le cadre des problèmes d'identification des paramètres libres d'un modèle. Dans ce cadre il faut résoudre des problèmes inverses et trop de paramètres libres peut faire devenir cette tâche impossible d'un point de vue du calcul. Le modèle à deux espèces de Darcy est capable de donner une description raisonnable des phénomènes. Non pas en occultant certain mécanismes biologiques, mais grâce à son nombre réduit de

paramètres libres par rapport à des modèles plus sophistiqués, il peut offrir un problème d'identification plus abordable.

Le modèle décrit l'écoulement saturé de trois phases dans un milieu isotope poreux non-uniforme. P, Q et S sont, respectivement, le nombre de cellules proliférantes (se divisant, responsables de la croissance tumorale), quiscentes et saines par unité de volume et les équations pour eux sont :

$$\frac{\partial P}{\partial t} + \nabla \cdot (\vec{v}P) = (2\gamma - 1)P + \gamma Q \tag{3.4}$$

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\vec{v}Q) = (\gamma - 1)P - \gamma Q \tag{3.5}$$

$$\frac{\partial S}{\partial t} + \nabla \cdot (\vec{v}S) = 0 \tag{3.6}$$

où la vitesse $\vec{v}$ explique la déformation du tissu et γ, une fonction scalaire des nutriments, détermine si les cellules tumorales prolifèrent ou meurent (en devenant quiescentes). On suppose un mouvement passif. On utilise la hypothèse d'écoulement saturé, $P + Q + S = 1$, et ce qui implique

$$\nabla \cdot \vec{v} = \gamma P \tag{3.7}$$

La fermeture mécanique du système est donnée par une loi de type Darcy pour la vitesse

$$\vec{v} = -k(P, Q)\nabla \Pi \tag{3.8}$$

où Π est la fonction scalaire jouant le rôle d'une pression (ou potentiel) et k est le champs de perméabilité donné par

$$k = k_1 + (k_2 - k_1)(P + Q) \tag{3.9}$$

étant k_1 la constante de perméabilité du tissu sain et k_2 la perméabilité du tissu tumorale. Les nutriments (en particulier, l'oxygène) sont gouvernés par une équation de diffusion-réaction:

$$\partial_t C - \nabla \cdot (D(P, Q)\nabla C) = 0.1(C_{max} - C)S - \alpha PC - 0.01\alpha QC \tag{3.10}$$

où C est la densité des nutriments, $C_{max} = 1$ et α est le taux de consommation des nutriments pour les cellules proliférantes.

$$D = D_{max} - K(P + Q) \tag{3.11}$$

est la diffusivité exprimée par une loi phénoménologique reflétant les différentes diffusions de l'oxygène dans le tissu sain et dans le tissu tumoral. La fonction γ exprime la transition entre l'état proliférant et l'état quiescent et c'est la régularisation de la fonction de Heaviside.

$$\gamma = \frac{1 + \tanh(R(C - C_{hyp}))}{2} \tag{3.12}$$

où R est un coefficient et C_{hyp} est la seuil d'hypoxie.

Le domaine Ω est généralement un domaine complexe avec un bord irrégulier. La masse ne peut pas sortir de ce domaine et, pour cette raison, des conditions

aux bords de Neumann homogènes sont imposées pour l'équation de la pression et pour l'équation de l'oxygène. Mais le problème de Neumann pour la pression doit être bien posé, il faut donc une modification de la divergence de la vitesse, qui doit être une quantité scalaire à moyenne nulle:

$$\nabla \cdot \vec{v} = \gamma(C) - \frac{\int_\Omega \gamma P \, d\Omega}{\int_\Omega 1 - P - Q \, d\Omega}(1 - P - Q). \tag{3.13}$$

Cette correction implique une compression du tissu sain causée par la croissance tumorale et donc l'équation (3.6) ne peut plus être valide.

Notre méthode est introduite pour résoudre l'équation pour la pression (3.8) et l'équation pour les nutriments (3.10) dans un domaine irrégulier. Le domaine original Ω est inséré dans un domaine plus grand et régulier. Les conditions aux bords originales sont considérées comme conditions de transmission sur l'interface séparant Ω et son complément relatif dans Ω', en introduisant de nouvelles diffusivité et perméabilité.

$$k' = \begin{cases} k, & \text{in } \Omega \\ k_{out}, & \text{in } \Omega' \setminus \Omega \end{cases} \tag{3.14}$$

$$D' = \begin{cases} D, & \text{in } \Omega \\ D_{out}, & \text{in } \Omega' \setminus \Omega \end{cases} \tag{3.15}$$

Dans l'esprit d'une méthode de pénalisation on impose des conditions aux bords de Neumann approximées pour la pression et pour l'oxygène sur le contour de Ω, en choisissant des très petites valeurs pour k_{out} et D_{out}.

On réalise deux simulations du modèle avec géométries et conditions initiales différentes. Le but est de montrer l'importance de la modélisation tridimensionnelle en croissance tumorale et le rôle joué par le bord de Ω dans l'évolution de la forme d'un nodule tumoral. Les interfaces (bords de Ω) choisies sont une sphère et un poumon. Le dernier est obtenu par segmentation d'images médicales (CT-scans). Les résultats fournissent, à un niveau qualitatif , un bon comportement de la méthode quant conditions de Neumann homogènes imposée sur le contour de Ω et une évolution raisonnable de la forme et de la composition du nodule.

Afin de corroborer ce qui est montré dans ce travail, il faudrait des études de la convergence de l'erreur avec interfaces tridimensionnelles plus complexes ainsi que, du coté simulation de croissance tumorale, des résultats quantitatifs et des comparaisons avec des cas réels. Gérer plus que une interface et donc plus que deux sous-domaines est une amélioration importante non seulement pour la méthode, mais également pour le modèle de croissance tumorale. Ceci introduirait la possibilité de considérer un domaine original structuré et donc plus vraisemblable. Ce sont les perspectives futures les plus importantes de ce travail.

3.3 Structure du travail

Dans le CHAPITRE 4 la méthode parallèle d'ordre deux pour problèmes elliptiques avec interface est introduite.

Dans la *section 4.1* une vue d'ensemble des méthodes existantes et de la méthode actuelle est donnée.
La *section 4.2* fournit la genèse de l'idée principale, en analysant l'erreur de convergence pour le problème unidimensionnel.
Dans la *section 4.3* le description détaillée de la méthode est illustrée pour le problème bidimensionnel. Ensuite, dans la *section 4.4*, l'approche dimension-par-dimension est utilisée afin de donner certains détails de la méthode pour le problème tridimensionnel.
L'implémentation parallèle est introduit dans la *section 4.5* avec des détails sur le modèle adopté, les outils utilisés et l'organisation du système linéaire.
La *section 4.6* présente la validation numérique de la méthode, en montrent les résultats sur le taux de convergence de l'erreur pour un bon ensemble de cas-test, en utilisant les deux implémentations, séquentielle et parallèle. Un cas-test tridimensionnel simple et les performances en scalabilité de l'implémentation parallèle sont discutés aussi dans cette section.
Les chapitre termine avec les conclusions sur la méthode dans la *section 4.7*.
Dans le CHAPITRE 5 la methode est appliqué pour resoudre le probleme elliptique dans un domaine irregulier dans le cadre d'un modèle de croissance tumorale.
Dans la *section 5.1* une brève introduction à la modélisation de croissance tumorale est donnée et le modèle choisi est illustré dans la *section 5.2*.
Ensuite, dans la *section 5.3* l'entier cadre numérique de l'application est introduit. On montre comment notre méthode peut être utilisée pour résoudre le problème elliptique dans un domaine irrégulier. Des notes sur les images médicales et les outils des segmentation sont fournis afin de montrer comment les géométries impliquées peuvent être obtenues.
Enfin, dans la *section 5.4* les résultats des simulations sont montrés et discutés.
Les conclusions sur l'application terminent le chapitre, dans la *section 5.5*.
Les perspectives et les intentions sur la méthode, sur le modèle de croissance tumorale et sur les applications alternatives sont discutées dans le CHAPTER 6.

Chapter 4

A parallel second order Cartesian method for elliptic interface problems, [28]

4.1 Introduction to the method

In the present work we aim to solve on Cartesian grids with an order two accuracy the following problem :

$$\nabla \cdot (k \nabla u) = f \text{ on } \Omega = \Omega_1 \cup \Omega_2 \tag{4.1}$$

$$[\![u]\!] = \alpha \text{ on } \Sigma \tag{4.2}$$

$$[\![k \frac{\partial u}{\partial n}]\!] = \beta \text{ on } \Sigma \tag{4.3}$$

assorted with boundary conditions on $\delta\Omega$ defined as the boundary of Ω, and where $[\![\cdot]\!]$ means $\cdot_1 - \cdot_2$. As illustrated on Figure 4.1, Ω consists in the union of two subdomains Ω_1 and Ω_2, separated by a complex interface Σ.

This elliptic problem with discontinuities across an interface appears in numerous physical or biological models. Among the well-known applications are heat transfer, electrostatics, fluid dynamics, but similar elliptic problems arise for instance in tumor growth modelling, where one has to solve a pressure equation [20], or in the modelling of electric potential in biological cells [21]. In this latter case the jump of the solution across the interface is proportional to the interior normal derivative.

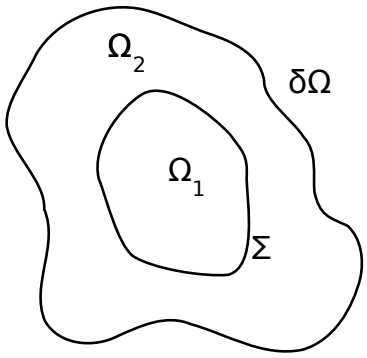


Figure 4.1: Geometry considered: two sub-domains Ω_1 and Ω_2 separated by a complex interface Σ

To solve an elliptic interface problem in the case of a complex interface, an alternative approach to body-fitted methods (see for instance [12], [19] and [26]) is to discretize and solve the problem on a Cartesian grid. In this case, one takes into account the influence of the complex interface through modifications of the numerical scheme near the interface, without the need of remeshing if the interface moves.

The first Cartesian grid method for elliptic problems was designed by Mayo in 1984 [66], and developed further in [67] and [68]. In that work an integral equation was derived to solve elliptic interface problems with piecewise coefficients to second order accuracy in maximum norm. Then LeVeque and Li (1994) [57] devised the very well known Immersed Interface Method (IIM). This

method relies on Taylor expansions of the solution on each side of the interface, with a local coordinate transformation near the interface to express the jump conditions in an appropriate frame. The elliptic operator is discretized on each grid point near the interface with formulas accounting for the jumps across the interface. In order to find these formulas a linear system with six unknowns needs to be solved for each of the concerned grid points. The method is also second order accurate in maximum norm. Numerous developments of the IIM have been performed. In the following lines we briefly evoke the most relevant ones. Li [58] developed a fast IIM algorithm for elliptic problems with piecewise constant coefficients. This version of IIM used auxiliary unknowns expressing the normal derivative at the interface. The fast IIM algorithm was generalized by Wiegmann and Bube in [89] under the name of Explicit Jump Immersed Interface Method (EJIIM). The EJIIM considers a classical finite difference discretization and uses corrective terms added to the right hand side of the linear system to take the interface into account. The corrective terms involve jumps of the solution and high order derivatives of it across the interface. Then Li and Ito [59] proposed to solve a quadratic optimization problem for each point near the interface in order to choose finite difference coefficients on a nine point stencil leading to a maximum principle preserving scheme (MIIM). Berthelsen devised the Decomposed Immersed Interface Method (DIIM) [17]. He used an iterative procedure to compute successive right hand side correction terms accounting for the jump conditions at the interface, associated to a nine point interpolation stencil on each side of the interface.

Another class of Cartesian method recently introduced by Zhou et al. is the Matched Interface and Boundary (MIB) method: [95], [94], [92]. This method can provide finite-difference schemes of arbitrary high order. The solution on each side of the interface is extended on fictitious points on the other side. These fictitious values are computed by iteratively enforcing the lowest order interface jump conditions. Finally, Chern and Shu [27] proposed a Coupling Interface Method, where the discretizations on each subdomain are coupled through a dimension by dimension approach using the jump conditions. All the methods cited above are second order accurate.

Other classes of Cartesian methods also exist, less accurate in the case of interface problems, but probably simpler to implement: Gibou et al. ([44], [45]) developed methods inspired by Fedkiw's Ghost-Fluid Method ([40], [39]) for multiphase flows. These methods are second order accurate for Dirichlet boundary conditions on arbitrary domains, but only first order accurate for interface problems. The AIIB method of Sarthou et al (submitted) is developed in the same spirit and it is second order accurate for Dirichlet boundary conditions and between order one and order two for interface problems. The penalty method, introduced by Arquis and Caltagirone [11] and Angot et al [9], consists in approximating fluid and solid by porous media with porosity tending respectively to infinity or zero. It can also be used to solve elliptic problems on arbitrary domains and is order one accurate. An improvement has been recently proposed by [23], to obtain second order accuracy for Dirichlet boundary conditions with an iterative scheme. But for the while it does not deal with interface problems.

All the methods cited before can be considered as finite differences methods, and this is the context in which we aim to present our method. However, Cartesian method for elliptic interface problems also exist in finite-volume and

finite-element communities: Collela and his group have notably devised methods in a finite volume spirit, where an interface reconstruction is applied to the cells near the interface (sometimes thus referenced as "cut cells"), in order to preserve conservativity properties ([54] and [69]). Among the finite-element community, let us give some references: [12], [19], [26], [52], [38], [53], [71] and [65].

To our knowledge, none of the second order Cartesian methods cited above has been implemented in a parallel code. One advantage of using Cartesian grids is to allow an easy parallelization, at least provided that the specific treatment of interfaces does not increase too much the complexity of the method. In this chapter we propose a parallel second order Cartesian method for elliptic interface problems, [28]. The method is based on a finite differences discretization and a dimension by dimension approach. In order to solve accurately the problem defined by equations (4.1)-(4.3) near the interface, we introduce additional unknowns located at the intersections of the interface with the grid. These interface unknowns are used in the discretization of the elliptic operator near the interface, and this avoid us to derive specific finite differences formulas containing jump terms, corrective terms, or to invert local linear systems, as in many other second order Cartesian methods. In order to solve for the interface unknowns we discretize and solve the flux jump conditions. The jump conditions and the coupling of the solution in the different subdomains are thus handled independently of the discretization of the elliptic equation, as it is the case in the MIB method, and the methods of Gibou and Sarthou too. But contrary to them, these additional unknowns are located at the interface and not at grid points of the other side of the interface. The simplicity of the method guarantees its easy parallelization.

4.2 Convergence rate dependence on truncation error for the one-dimensional problem

In this section, we consider the Laplacian equation in 1D and we analyse the convergence error in terms of the truncation error. This will allow us to explain the choice of a discretization near the interface in order to get a second order accuracy.

First of all, let us point out that the influence of the truncation error in a given point on the convergence of the scheme depends of the location of this point. We assume that a boundary or an interface lies between grid points i and $i+1$. To approximate the Laplacian operator at point i we consider a ghost-cell formula. We create a fictitious value at point $i+1$ taking into account the boundary or interface condition $u(x_{int}) = u_{int}$. u_{int} may be a fixed value if we consider a Dirichlet boundary condition or may depend on the solution on the other side of the interface if we consider interface flux transmissions. If we use a linear extrapolation the ghost value at point $i+1$will be:

$$\tilde{u}_{i+1} = u_i + dx\frac{u_{int} - u_i}{x_{int} - x_i} \tag{4.4}$$

The truncation error ϵ_i of the discretization of the Laplacian at point i is therefore:

$$\epsilon_i = \frac{dx\frac{(x_{int}-x_i)}{2}u''(x_i) + \frac{dx^2}{2}u''(x_i) + O(dx^3)}{dx^2} = O(1) \tag{4.5}$$

and the scheme is not formally consistent in the finite differences sense at point i.

In the case of boundary conditions, theoretical studies performed by Gustafsson ([48] and [49]) and more recently by Svard and Nordstrom [86] show that under certain assumptions a discretization less accurate at the boundary than in the rest of the domain does not deteriorate the order of convergence. In fact the numerical methods developed in the ghost-cell spirit ([44], [45]) are practically at least second order accurate for Dirichlet boundary conditions. However, in the case of immersed interface problems, only first order convergence is reached for ghost cell methods, like for instance in [61] and [81]. For this reason, we study in a simple case the influence of the truncation error located on the interface.

We consider the one-dimensional Laplace equation on the segment $[0, 1]$, with Dirichlet boundary conditions. The following jump conditions are satisfied at the interface Σ located at $x = x_{int}$.

$$[\![u]\!] = 0 \text{ on } \Sigma \tag{4.6}$$

$$[\![k\frac{\partial u}{\partial n}]\!] = 0 \text{ on } \Sigma \tag{4.7}$$

We assume that $k = k'$ in $[0, x_{int}]$ and $k = k''$ in $]x_{int}, 1]$. Grid points are defined on locations $x_m = mdx, 0 \leq m \leq N + 1$ with $dx = \frac{1}{N+1}$ and x_{int} belongs to the segment $]x_i, x_{i+1}[$. The interface stands inside the domain so we can write $i \sim aN$, with a a real between 0 and 1, independent of dx. For the grid points inside the domain and far enough from the interface, we use the second order discretization of equation 4.1:

$$\frac{u_{m+1} - 2u_m + u_{m-1}}{dx^2} = f_m \tag{4.8}$$

with f_m the value of f at point x_m.

The local error $e_m = u(x_m) - u_m$ satisfies the same linear relationship as u_k with the local truncation error ϵ_m as a source term:

$$\frac{e_{m+1} - 2e_m + e_{m-1}}{dx^2} = \epsilon_m. \tag{4.9}$$

Now we assume that a Ghost-Cell technique based on a linear extrapolation is used to discretize the Laplacian near the interface. Let us denote u_{int} the numerical solution at the interface. We obtain the following finite differences formulas for the discretization of the Laplace equation at points i and $i + 1$:

$$\frac{u_{int} - u_i}{x_{int} - x_i} - \frac{u_i - u_{i-1}}{dx} = dx f_i \tag{4.10}$$

$$\frac{u_{i+2} - u_{i+1}}{dx} - \frac{u_{i+1} - u_{int}}{x_{i+1} - x_{int}} - = dx f_{i+1} \tag{4.11}$$

Then the truncation errors ϵ_i and ϵ_{i+1} on points i and $i + 1$ satisfy:

$$\frac{e_{int} - e_i}{x_{int} - x_i} - \frac{e_j - e_{i-1}}{dx} = dx\ \epsilon_i \tag{4.12}$$

$$\frac{e_{i+2} - e_{i+1}}{dx} - \frac{e_{i+1} - e_{int}}{x_{i+1} - x_{int}} = dx\ \epsilon_{i+1} \tag{4.13}$$

with $e_{int} = u(x_{int}) - u_{int}$.

The flux equation at interface is expressed by another linear approximation:

$$k_{i+1}\frac{u_{i+1}-u_{int}}{x_{i+1}-x_{int}}- \quad = \quad k_i\frac{u_{int}-u_i}{x_{int}-x_i} \tag{4.14}$$

with k_i the value of coefficient k at point x_i. Thus the truncation error ϵ_{int} related to this equation satisfies:

$$k_{i+1}\frac{e_{i+1}-e_{int}}{x_{i+1}-x_{int}} - k_i\frac{e_{int}-e_i}{x_{int}-x_i} = \epsilon_{int} \tag{4.15}$$

The truncation error of the Laplacian discretizations (4.10) and (4.11) are zero order accurate, and the truncation error of the flux equation (4.14) is order one accurate, as a Taylor expansion would show in the same way as in (4.5) .

In order to close the linear system, we assume that Dirichlet boundary conditions are imposed exactly:

$$e_0 \quad = \quad 0 \text{ and } e_{N+1} = 0. \tag{4.16}$$

We aim to solve explicitly the linear system satisfied by the truncation error. By two recurrences, one forward and one backward, we can show that:

$$e_{m+n} \quad = \quad (n+1)e_m - ne_{m-1} + dx^2\sum_{j=1}^{n} j\epsilon_{m+n-j} \tag{4.17}$$

$$e_{m-n} \quad = \quad (n+1)e_m - ne_{m+1} + dx^2\sum_{j=1}^{n} j\epsilon_{m-n+j} \tag{4.18}$$

Using the boundary conditions, we deduce from (4.17) and (4.18) that:

$$0 = (N+1-m)e_{m-1} - (N+2-m)e_m - dx^2\sum_{j=1}^{N+1-m} j\epsilon_{N+1-j} \qquad \text{for } m \geq i+2 \tag{4.19}$$

$$0 = (m+1)e_m - me_{m+1} + dx^2\sum_{j=1}^{m} j\epsilon_j \qquad \text{for } m \leq i-1 \tag{4.20}$$

Therefore:

$$e_{m+1} = \frac{(N-m)e_m}{(N+1-m)} - dx^2\sum_{j=1}^{N-m}\frac{j}{(N+1-m)}\epsilon_{N+1-j} \qquad \text{for } m \geq i+2 \tag{4.21}$$

$$e_{m+1} = \frac{(m+1)}{m}e_m + dx^2\sum_{j=1}^{m}\frac{j}{m}\epsilon_j \qquad \text{for } m \leq i-1 \tag{4.22}$$

Using the latter equations with (4.12), (4.13) and (4.15) we show that:

$$e_{int}(C_i + C_{i+1}) = \quad -\epsilon_{int} - dx^2\ C_i\ \sum_{j=1}^{N-i} j\ \epsilon_{N+1-j} - dx^2\ C_{i+1}\ \sum_{j=1}^{i} j\ \epsilon_j \tag{4.23}$$

with:

$$C_i = \frac{k_1}{x_{int} - x_i + idx} \tag{4.24}$$

$$C_{i+1} = \frac{k_2}{x_{i+1} - x_{int} + (N-i)dx} \tag{4.25}$$

Because $i \sim aN$ then $C_i = O(1)$ and $C_{i+1} = O(1)$. Therefore if the truncation error of the flux equation ϵ_{int} is order one then the error on the interface e_{int} is a priori order one. Moreover, if the truncation error of the Laplacian is order zero near the interface: $\epsilon_i = O(1)$ and $\epsilon_{i+1} = O(1)$, then the error on the interface point is a priori order one too. We deduce that to obtain an order 2 accuracy at the interface, we need to:

- use a discretization of Laplacian near the interface with a truncation error of order 1, thus avoid linear extrapolations.
- use a discretization of the flux transmission equations at the interface with a truncation error of order 2.

This analysis is not a proof that if the truncation error of the Laplacian is zero, then the numerical error will never be second order accurate, because there can be compensation effects in the sums in the expression of the error. However numerical results with Ghost-cell like methods in [61] and [81] corroborate this reasoning.

4.3 Description of the method for the two-dimensional problem

In this section we firstly describe the method in the case $\alpha = \beta = 0$, where α and β are the jumps of the solution and its co-normal derivative, as presented in the equations (4.2) and (4.3). The case $\alpha \neq 0$, $\beta \neq 0$ will be treated in the last part of the section.

4.3.1 Interface description and classification of grid points

In order to improve accuracy in the vicinity of the interface we need additional geometric information. This information, mainly the distance from the interface and the normal to the interface, can be provided by the level set method, introduced by Osher and Sethian [73]. We refer the interested reader to [83], [85] and [72] for recent reviews of this method. The zero isoline of the level set function, defined by:

$$\varphi(x) = \begin{cases} dist_\Sigma(x) & \text{outside of the interface} \\ -dist_\Sigma(x) & \text{inside of the interface} \end{cases} \tag{4.26}$$

represents implicitly the interface Σ immersed in the computational domain. The scalar function $dist_\Sigma$ gives the Euclidean distance between its argument and Σ.

A useful property of the level set function is:

$$\mathbf{n}(x) = \nabla\varphi(x) \tag{4.27}$$

where $\mathbf{n}(x)$ is the outward normal vector of the isoline of ϕ passing through x. In particular, this allows to compute the values of the normals to the interface.

The points on the Cartesian grid are defined by $M_{i,j} = (x_i, y_j) = (i\,dx, j\,dy)$. We denote by $u_{i,j}$ the approximation of u at the point (x_i, y_j). We say that a grid point is neighbouring the interface if ϕ changes sign between this point and at least one of its neighbours. In the contrary, regular grid points designate grid points that are not neighbours of the interface. Figure 4.2 gives an example of this classification.

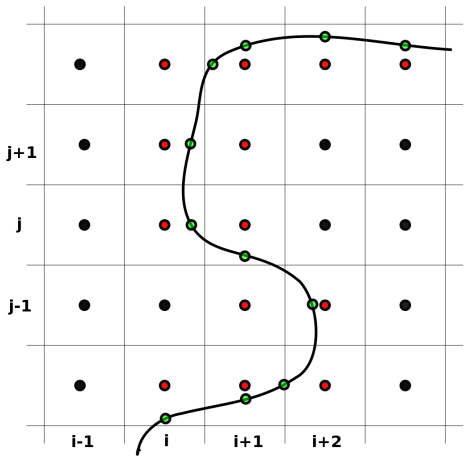


Figure 4.2: Grid points classification: regular grid points are in full black, interface grid points are in red-black and intersection points are in green-black. Straight lines mark cells off.

4.3.2 Discrete elliptic operator for regular grid points

On regular grid points we use the standard centred second order finite differences scheme to approximate 4.1:

$$\nabla \cdot (k\nabla u)(x_i, y_j) \approx \frac{k_{i+1/2,j}(u_{i+1,j} - u_{i,j}) - k_{i-1/2,j}(u_{i,j} - u_{i-1,j})}{dx^2} + \\ + \frac{k_{i,j+1/2}(u_{i,j+1} - u_{i,j}) - k_{i,j-1/2}(u_{i,j} - u_{i,j-1})}{dy^2} \quad (4.28)$$

with $k_{i+1/2,j}$ a second order approximation of k at point $\frac{M_{i,j}+M_{i+1,j}}{2}$.

4.3.3 Discrete elliptic operator near the interface

On grid points neighbouring the interface, the latter approximation is not accurate enough, because of the discontinuity of the coefficient k. In order to discretize more accurately the term $\nabla(k\nabla u)(x_i, y_j)$, we create new unknowns, located at what we call "intersection points". The intersection points are defined the following way: if the intersection of the interface and $[M_{i,j}M_{i+1,j}]$ exists, then we define the intersection point $I_{i+1/2,j} = (\tilde{x}_{i+1/2}, y_j)$ as this intersection. The intersection point $I_{i,j+1/2} = (x_i, \tilde{y}_{j+1/2})$ is similarly defined as

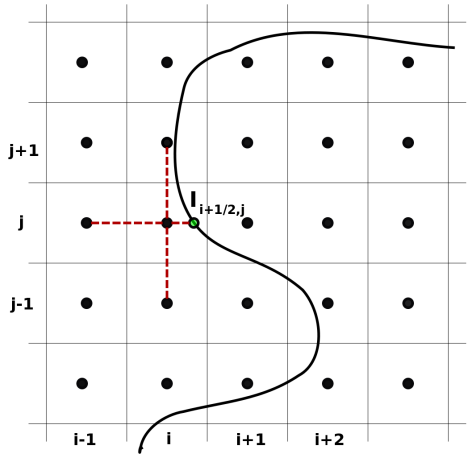


Figure 4.3: Example of stencil for the discretization of the elliptic operator

the intersection, if it exists, of the interface and the segment $[M_{i,j}M_{i,j+1}]$. See Figure 4.2.

For a grid point $M_{i,j}$ neighbouring the interface, we discretize the term of equation (4.1) $\nabla \cdot (k\nabla u)(x_i, y_j)$ with the values on $M_{i,j}$ and the closest points (grid or intersection points) to $M_{i,j}$ in each direction. For instance, in the case illustrated on Figure 4.3, we get:

$$\begin{aligned}\nabla \cdot (k\nabla u)(x_i, y_j) =& \frac{\tilde{k}_{i+1/2,j}\frac{\tilde{u}_{i+1/2,j}-u_{i,j}}{\tilde{x}_{i+1/2}-x_i} - k_{i-1/2,j}\frac{u_{i,j}-u_{i-1,j}}{dx}}{\frac{(\tilde{x}_{i+1/2}-x_i)}{2}+\frac{dx}{2}} \\ &+ \frac{k_{i,j+1/2}(u_{i,j+1}-u_{i,j}) - k_{i,j-1/2}(u_{i,j}-u_{i,j-1})}{dy^2}\end{aligned} \tag{4.29}$$

where $\tilde{u}_{i+1/2,j}$ denotes the value of u at point $I_{i+1/2,j}$, and $\tilde{k}_{i+1/2,j}$ is an approximation of the value of k at the middle between $I_{i+1/2,j}$ and $M_{i,j}$. This discretization is second order accurate if $\tilde{x}_{i-1/2,j}$ coincides with x_{i+1}, and first order otherwise.

4.3.4 Discrete flux transmission conditions

As we have seen in the last section, we want the truncation error of the discretization of flux equality (4.3) to be second order accurate. On Figure 4.4 a possible configuration of the interface is illustrated. In the x-direction, it is straightforward to compute a second order approximation of the x-derivative with three non equidistant points. For example, with the points $M_{i-1,j}$, $M_{i,j}$ and $I_{i+1/2,j}$, we have:

$$\begin{aligned}\frac{\partial u^l}{\partial x}(\tilde{x}_{i+1/2}, y_j) \approx& \frac{(u_{i-1,j}-\tilde{u}_{i+1/2,j})(x_i-\tilde{x}_{i+1/2})}{dx(x_{i-1}-\tilde{x}_{i+1/2})} + \\ &- \frac{(u_{i,j}-\tilde{u}_{i+1/2,j})(x_{i-1}-\tilde{x}_{i+1/2})}{dx(x_i-\tilde{x}_{i+1/2})}\end{aligned} \tag{4.30}$$

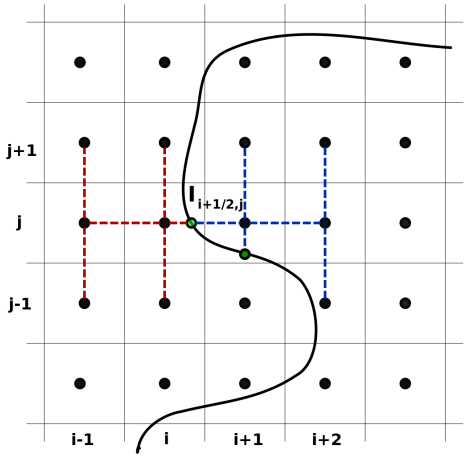


Figure 4.4: Example of order two flux discretization at point $I_{i+1/2,j}$.

The right x-derivative $\frac{\partial u^r}{\partial x}(\tilde{x}_{i+1/2,j}, y_j)$ is approximated in the same way. For the derivative along the y-direction, we do not have unknowns located on the line parallel to the y-axis passing by $I_{i+1/2,j}$. Therefore we use an extended stencil including 6 points. We use a linear combination of $(\frac{\partial u}{\partial y})_{i,j}$ and $(\frac{\partial u}{\partial y})_{i-1,j}$, defined respectively as second order approximations of the y-derivative on $M_{i,j}$ and $M_{i-1,j}$ to approximate the derivative on $I_{i+1/2,j}$ with second order accuracy:

$$\frac{\partial u^l}{\partial y}(\tilde{x}_{i+1/2}, y_j) \approx \quad \frac{\tilde{x}_{i+1/2} - x_{i-1}}{dx} (\frac{\partial u}{\partial y})_{i,j} - \frac{\tilde{x}_{i+1/2} - x_i}{dx} (\frac{\partial u}{\partial y})_{i-1,j} \qquad (4.31)$$

The formulas for $(\frac{\partial u}{\partial y})_{i,j}$ and $(\frac{\partial u}{\partial y})_{i-1,j}$ depend on the local configuration on the interface, but they are based on the same principle as for (4.30). The formulas (4.30) and (4.31) are consistent if the point $M_{i-1,j}$ belongs to the same domain as $M_{i,j}$. We thus need that there are at least two adjacent points in each

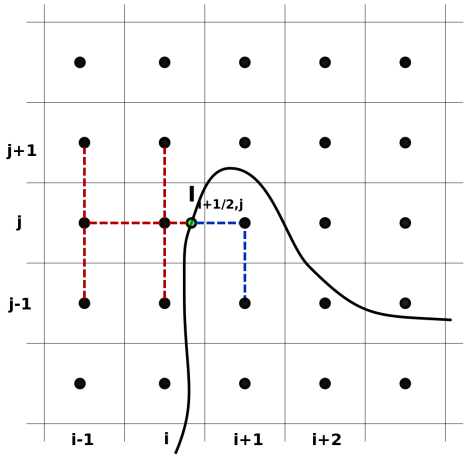


Figure 4.5: Example of order one flux discretization at point $I_{i+1/2,j}$.

direction belonging to the same domain. If on one side of the interface the two closest grid points aligned with the intersection point do not belong to the same sub-domain, the second order discretization is not possible any more. In this case, we use instead for the problematic side of the interface a first order discretization involving only three points. Such a case is illustrated on Figure 4.5.

Finally the flux equality (4.3) corresponding to the cases described in Figures 4.4-4.5 is discretized by:

$$[k^l(\frac{\partial u^l}{\partial x})_{i+1/2,j} - k^r(\frac{\partial u^r}{\partial x})_{i+1/2,j}]\ n_x + [k^l(\frac{\partial u^l}{\partial y})_{i+1/2,j} - k^r(\frac{\partial u^r}{\partial y})_{i+1/2,j}]\ n_y = 0 \tag{4.32}$$

where (n_x, n_y) is an approximation of the vector normal to the interface at point $I_{i+1/2,j}$, k^l and k^r respectively the left and right limit values of k at $I_{i+1/2,j}$.

The stencil used in this discretization of the flux equality contains 13 points. Actually, it is possible to use only 8 points. As other authors noticed ([89], [17], [95] and [27]) the jump condition on u can be differentiated in the direction tangential to the interface:

$$[\![\frac{\partial u}{\partial \tau}]\!] \quad = \quad \frac{\partial \alpha}{\partial \tau} \tag{4.33}$$

The latter equation is a linear relationship between the partial derivatives on each side on the interface: $\frac{\partial u^l}{\partial x}$, $\frac{\partial u^r}{\partial x}$, $\frac{\partial u^l}{\partial y}$ and $\frac{\partial u^r}{\partial y}$. For instance $\frac{\partial u^r}{\partial y}$ can be expressed as a linear combination of the others partial derivatives, and does not need to be discretized, removing 4 points from the stencil used for the discretization of the flux. Additionally, one can use only 5 points instead of 6 to discretize $\frac{\partial u^l}{\partial y}$ with second order accuracy. However, when we compared the 13 points and the 8 points versions, we noticed that the amplitude of the error was sensibly higher when using the 8 points stencil, while the computational time was almost the same. This observation is illustrated for Problem 3 of the numerical validation, in section 4.6 .

4.3.5 Stabilization

In our first numerical tests, we noticed oscillations in the convergence plots. These instabilities appeared when the discretization of a flux equation at an interface point involved another intersection point located very close to a grid point. This situation is illustrated on Figure 4.6. In this figure one can understand why such a configuration leads to numerical instabilities: the finite differences formula creates a coupling between the two interface unknowns and it is ill-conditioned because of the small distance between the grid point and the intersection point.

To avoid these oscillations we use in these cases a de-centred discretization. As illustrated in Figure 4.7, instead of using in the flux discretization the second intersection point, we use the closest grid point located in the opposite direction.

For instance, in Eq. (4.38) the term $(\frac{\partial u^r}{\partial y})_{i+1/2,j}$ is computed with a second order finite differences formula involving the grid points $(i+1, j)$, $(i+1, j+1)$ and $(i+1, j+2)$ instead of the grid points $(i+1, j)$, $(i+1, j+1)$ and the intersection point $I_{i+1,j-1/2}$. We decided to use the modified stencil in a systematic way

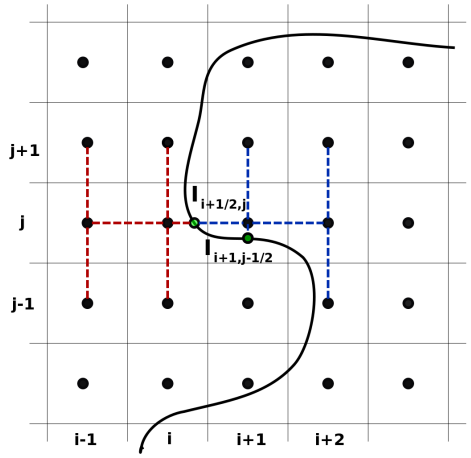


Figure 4.6: Centred stencil: the discretization of the flux on point $I_{i+1/2,j}$ involves intersection point $I_{i+1,j-1/2}$ and it is not numerically stable.

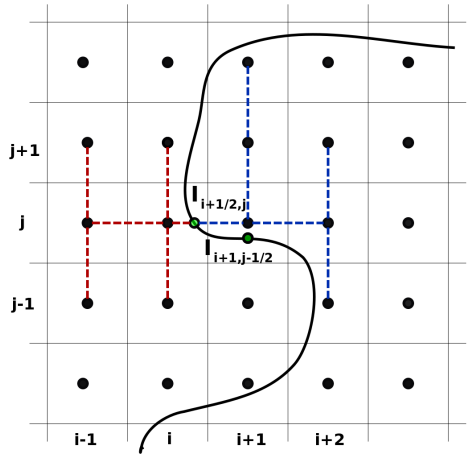


Figure 4.7: Decentered stencil: the discretization of the flux on point $I_{i+1/2,j}$ involves grid point $(i+1, j+2)$ instead of intersection point $I_{i+1,j-1/2}$

every time there are two intersection points involved in the same flux equation. The effect of the stabilization is illustrated for Problem 1 of the numerical validation in subsection 4.6.

4.3.6 Case $\alpha \neq 0$, $\beta \neq 0$

Here non-zero jumps of the function and of its normal derivative across the interface are taken into account. The jump term α being non-zero means that there are in fact two unknowns at each intersection point. We define as interface unknowns at the points $I_{i+1/2,j}$ or $I_{i,j+1/2}$ the limit values of u in the domain defined by $\phi < 0$: $\tilde{u}_{i+1/2,j}$ or $\tilde{u}_{i,j+1/2}$. The values at the same points but for the domain where $\phi > 0$ are defined by $\tilde{u}_{i+1/2,j} + \alpha(I_{i+1/2,j})$ or $\tilde{u}_{i,j+1/2} + \alpha(I_{i,j+1/2})$. Only the right hand side term of the linear system changes. For the

lines of the linear system involving the interface unknowns of the sub-domain $\phi > 0$, the right hand side term receives the value $-\alpha(I_{i+1/2,j})$ or $-\alpha(I_{i,j+1/2})$ multiplied by the coefficient used for the interface unknowns $\tilde{u}_{i+1/2,j}$ or $\tilde{u}_{i,j+1/2}$ themselves. For instance, if $\varphi(M_{i,j}) < 0$ then Eq. (4.29) remains unchanged, but if $\varphi(M_{i,j}) > 0$ it becomes:

$$\begin{aligned}
&\frac{\tilde{k}_{i+1/2,j}\frac{\tilde{u}_{i+1/2,j}-u_{i,j}}{\tilde{x}_{i+1/2}-x_i} - k_{i-1/2,j}\frac{u_{i,j}-u_{i-1,j}}{dx}}{\frac{(\tilde{x}_{i+1/2}-x_i)}{2}+\frac{dx}{2}}+ \\
&+\frac{k_{i,j+1/2}(u_{i,j+1}-u_{i,j}) - k_{i,j-1/2}(u_{i,j}-u_{i,j-1})}{dy^2} = \\
&= -\frac{\frac{\tilde{k}_{i+1/2,j}}{(\tilde{x}_{i+1/2}-x_i)}}{\frac{(\tilde{x}_{i+1/2}-x_i)}{2}+\frac{dx}{2}}\alpha(I_{i+1/2,j}) + f(x_i,y_j)
\end{aligned} \tag{4.34}$$

If the jump term β is non-zero, the coefficient $\beta(I_{i+1/2,j})$ or $\beta(I_{i,j+1/2})$ appears in the right hand side of the equation discretizing the flux equality on point $I_{i+1/2,j}$ or $I_{i,j+1/2}$. For instance, if $\varphi(M_{i,j}) > 0$ Eq. (4.38) becomes:

$$\begin{aligned}
&[k_{i,j}(\frac{\partial u^l}{\partial x})_{i+1/2,j} - k_{i+1,j}(\frac{\partial u^r}{\partial x})_{i+1/2,j}]\ n_x + \\
&+[k_{i,j}(\frac{\partial u^l}{\partial y})_{i+1/2,j} - k_{i+1,j}(\frac{\partial u^r}{\partial y})_{i+1/2,j}]\ n_y = \beta(I_{i+1/2,j})
\end{aligned} \tag{4.35}$$

4.4 Three-dimensional extension by dimensional splitting

The dimension-by-dimension approach allows a straightforward extension of the method to the three-dimensional problem. In this section the finite differences approximation for equations (4.1)-(4.3) is given for $\alpha = 0$ and $\beta = 0$. Points classification follows the same principle as in the two-dimensional case and the information about the interface is still given by the level set function.

The points on the Cartesian grid are now defined by $M_{i,j,k} = (x_i, y_j, z_k) = (idx, jdy, kdz)$. $u_{i,j,k}$ denotes the approximation of u at $M_{i,j,k}$. On regular grid points the standard centred second order finite differences scheme approximating the left-hand side of the equation (4.1) reads:

$$\begin{aligned}
\nabla\cdot(k\nabla u)(x_i,y_j,z_k) \approx &\frac{k_{i+1/2,j,k}(u_{i+1,j,k}-u_{i,j,k}) - k_{i-1/2,j,k}(u_{i,j,k}-u_{i-1,j,k})}{dx^2}+ \\
&+\frac{k_{i,j+1/2,k}(u_{i,j+1,k}-u_{i,j,k}) - k_{i,j-1/2,k}(u_{i,j,k}-u_{i,j-1,k})}{dy^2}+ \\
&+\frac{k_{i,j,k+1/2}(u_{i,j,k+1}-u_{i,j,k}) - k_{i,j,k-1/2}(u_{i,j,k}-u_{i,j,k-1})}{dz^2}
\end{aligned} \tag{4.36}$$

with $k_{i,j,k+1/2}$ a second order approximation of k at point $\frac{M_{i,j,k}+M_{i,j,k+1}}{2}$.

Considering the presence of an intersection, $I_{i+1/2,j,k}$, in $]M_{i,j,k}, M_{i+1,j,k}[$ (see Figure 4.8), for the point $M_{i,j,k}$ close to the interface, the finite differences

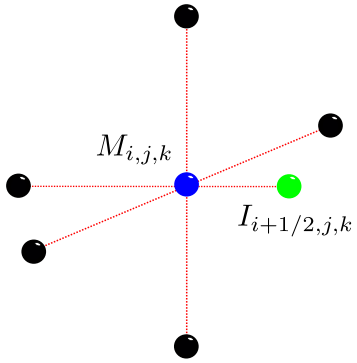


Figure 4.8: Example of 3D elliptic operator stencil near the interface.

scheme in equation (4.36) changes, as the one in equation (4.28) changes in (4.29); in the case of an intersection in the x-direction:

$$\nabla \cdot (k\nabla u)(x_i, y_j, z_k) \approx \frac{\tilde{k}_{i+1/2,j,k}\frac{\tilde{u}_{i+1/2,j,k}-u_{i,j,k}}{\tilde{x}_{i+1/2}-x_i} - k_{i-1/2,j,k}\frac{u_{i,j,k}-u_{i-1,j,k}}{dx}}{\frac{(\tilde{x}_{i+1/2}-x_i)}{2}+\frac{dx}{2}} + \\ + \frac{k_{i,j+1/2,k}(u_{i,j+1,k}-u_{i,j,k}) - k_{i,j-1/2,k}(u_{i,j,k}-u_{i,j-1,k})}{dy^2} + \\ + \frac{k_{i,j,k+1/2}(u_{i,j,k+1}-u_{i,j,k}) - k_{i,j,k-1/2}(u_{i,j,k}-u_{i,j,k-1})}{dz^2} \tag{4.37}$$

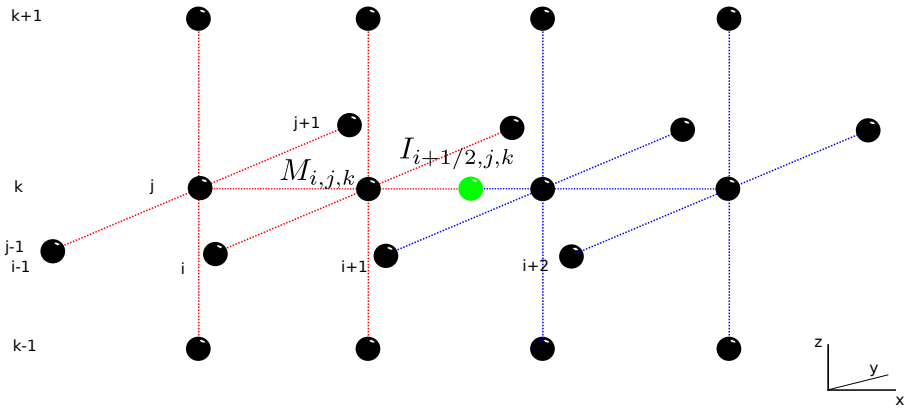


Figure 4.9: Example of equation (4.3) stencil for the three-dimensional problem.

The co-normal jump condition now reads (for $\beta = 0$), for the situation in Figure 4.9:

$$\begin{aligned}
&[k^l(\frac{\partial u^l}{\partial x})_{i+1/2,j,k} - k^r(\frac{\partial u^r}{\partial x})_{i+1/2,j,k}]\ n_x + \\
&+ [k^l(\frac{\partial u^l}{\partial y})_{i+1/2,j,k} - k^r(\frac{\partial u^r}{\partial y})_{i+1/2,j,k}]\ n_y + \\
&+ [k^l(\frac{\partial u^l}{\partial z})_{i+1/2,j,k} - k^r(\frac{\partial u^r}{\partial z})_{i+1/2,j,k}]\ n_z = 0
\end{aligned} \tag{4.38}$$

where, considering the left side relative to the intersection (the red side in Figure 4.9):

$$\begin{aligned}
\frac{\partial u^l}{\partial x}(\tilde{x}_{i+1/2}, y_j, z_k) \approx & \frac{(u_{i-1,j,k} - \tilde{u}_{i+1/2,j,k})(x_i - \tilde{x}_{i+1/2})}{dx(x_{i-1} - \tilde{x}_{i+1/2})} + \\
& - \frac{(u_{i,j,k} - \tilde{u}_{i+1/2,j,k})(x_{i-1} - \tilde{x}_{i+1/2})}{dx(x_i - \tilde{x}_{i+1/2})}
\end{aligned} \tag{4.39}$$

and

$$\frac{\partial u^l}{\partial y}(\tilde{x}_{i+1/2}, y_j, z_k) \approx \quad \frac{\tilde{x}_{i+1/2} - x_{i-1}}{dx}\ (\frac{\partial u}{\partial y})_{i,j,k} - \frac{\tilde{x}_{i+1/2} - x_i}{dx}\ (\frac{\partial u}{\partial y})_{i-1,j,k} \tag{4.40}$$

$$\frac{\partial u^l}{\partial z}(\tilde{x}_{i+1/2}, y_j, z_k) \approx \quad \frac{\tilde{x}_{i+1/2} - x_{i-1}}{dx}\ (\frac{\partial u}{\partial z})_{i,j,k} - \frac{\tilde{x}_{i+1/2} - x_i}{dx}\ (\frac{\partial u}{\partial z})_{i-1,j,k} \tag{4.41}$$

The same considerations for the two-dimensional case still hold: the right side derivatives are in the same spirit of equations (4.39)-(4.41); derivatives appearing in (4.40) and (4.41) at points i and $i-1$ still depends on interface topology, however, they are based on the same principle of (4.39), as in the two-dimensional problem. The stabilization follows the same two-dimensional procedure too, shifting the scheme position of one or more derivatives (in this case we can have up to two too close intersections for each side) in order to avoid the presence of too close intersections in the stencil, exactly with the same spirit introduced before. First order discretization is mandatory if less than two points aligned with the intersection are in the same sub-domain. Examples of stabilization and first order discretization for the three-dimensional case are given in Figures 4.10 and 4.11. Nothing more than the third direction is needed to handle the three-dimensional problem. All the possible intersections and interface topologies can be properly managed with the ingredients in this section.

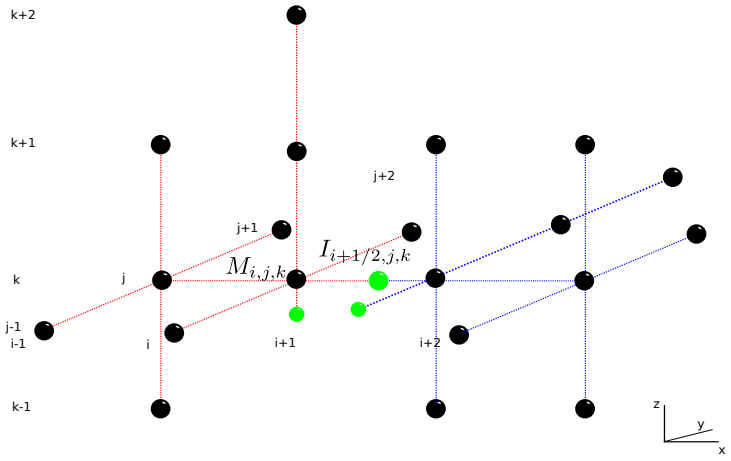


Figure 4.10: Example of stabilization for the three-dimensional problem. Smaller green balls are too close intersections for $I_{i+1/2,j,k}$

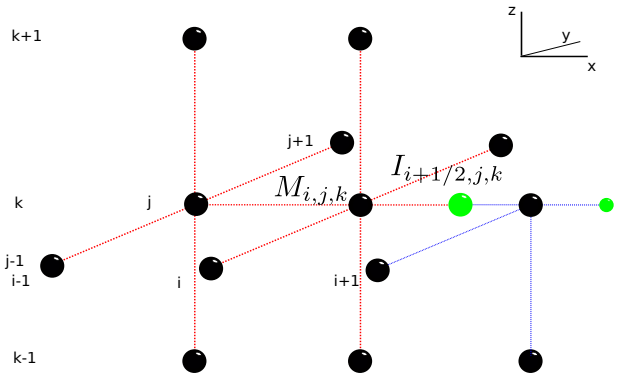


Figure 4.11: Example of order one flux discretization in $I_{i+1/2,j,k}$ for the three-dimensional problem. Smaller green ball indicates the intersection avoiding the order two discretization.

4.5 Parallelization of the method

Growing in interface topology complexity involves the need for a large number of points in order to catch the near interface details of the solution. Moreover, if the method is employed in the frame of time integration methods, which need to solve an immersed interface elliptic problem at every time step, the efficiency and the performance of the solver become crucial. The parallel implementation of the method allows to deal with both the matters.

4.5.1 Parallelization model and PETSc library

The parallelization of the method has been handled using the local memory parallel programming paradigm, Message Passing, and the SPMD philosophy

(Single Program Multiple Data). The API (Application Program Interface) chosen to implement the former is MPI (Message Passing Interface, [2]).

The implemented parallelization model, the parallelism of the data, is typical of problems coming from Partial Differential Equations, because it allows a good scalability compared to the functional one. The computational domain is decomposed and every single part is assigned to a processor which, then, executes the program on its sub-domain.

The management of the sub-domains boundaries is crucial for a good efficiency of the parallel code, in particular if the values of the solution in the points of the sub-domain near to the boundary depend on points belonging to the adjacent sub-domain. The most common approach to the boundary management is extending the sub-domain of a processor to points of the adjacent sub-domain. These points make up a new area of the single processor computational domain, called Ghost Region. The values of the solution in this region are not modified by the processor they belong to, as Ghost Points, but they are updated through communications with contiguous processors. The Cartesian topology has been chosen in analogy to the computational grid and it ensures, at least in the general cases, the smallest amount of communications to update the Ghost Regions.

Using the PETSc library (Portable, Extensible Toolkit for Scientific Computation, see [14], [13], [15] for details) we were able to avoid explicit communications. PETSc supplies the user with parallel data structures (vectors, matrices, index sets and more), efficient access and assembling operations for these structures, and in particular several iterative linear and non-linear solvers.

4.5.2 Parallel implementation

The aim of the code is to solve the following linear system

$$Au = f \tag{4.42}$$

where A is the matrix discretizing the differential operator on the Cartesian grid points and the transmission conditions on the interface, according to the equations in Sections 4.3.2-4.3.4; f is the right hand side on the grid points augmented with the jump values of the fluxes on the interface points. The solution is stored in u, which contains the values on the grid points and on the interface points. Therefore, the code searches the intersections, assembles A and f, solves the linear system and extracts the solution at the grid points.

The matrix

In this section we want to sketch how the presence of the interface and the PETSc parallelization define the matrix sparsity pattern and how the matrix is partitioned among the processors.

The matrix non-zeros pattern strongly depends on the position of the interface in the computational domain and on the grid points enumeration introduced by PETSc. Figure 4.13b shows the local and global enumeration of grid points and intersections. Without loss of generality we want to show what happen in the matrix for points near to the interface Figure 4.12 illustrates the matrix structure in the case of Figure 4.13b for grid points $0 \leq i \leq 9$ and $2 \leq j \leq 3$

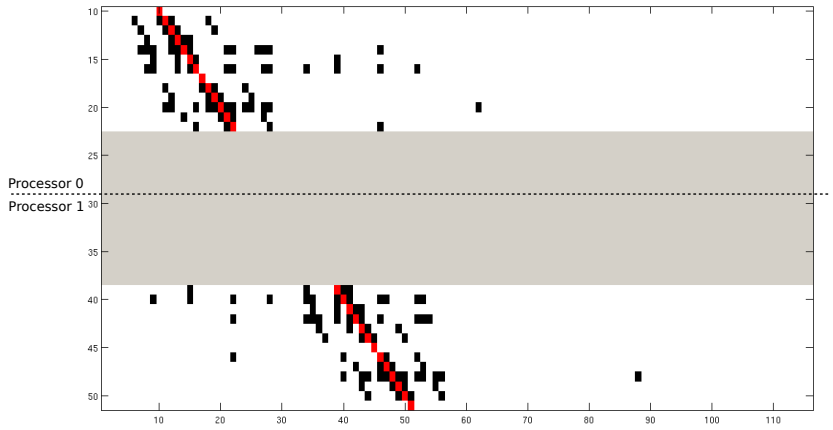


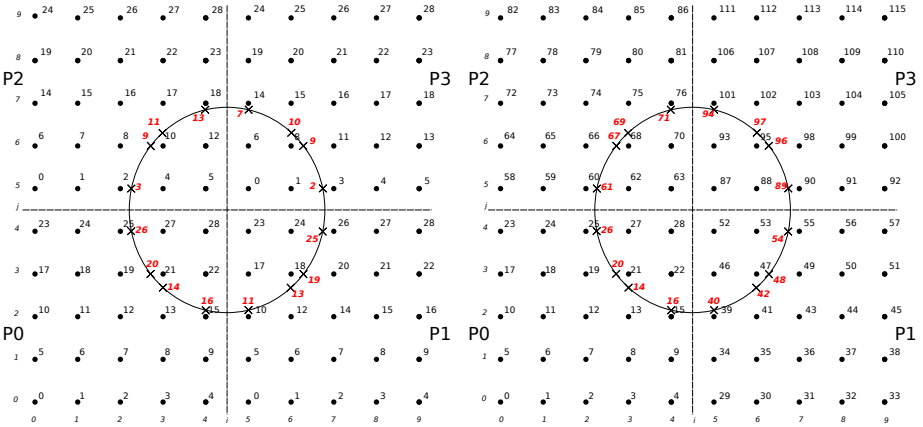
Figure 4.12: A-matrix rows relative to points (i, j) in the ranges $0 \leq i \leq 9$ and $2 \leq j \leq 3$, in Figure 4.13b. White means zero valued matrix elements, black and red mean non-zeros (red for elements on the principal diagonal). Grey is for omitted rows (corresponding to points outside the considered ranges). The separation between Processor 0 and Processor 1 is shown.

and intersection points in the same area, in the case of Dirichlet boundary conditions.

The matrix is row partitioned: a number of rows equal to the sum of grid points and intersections in its own sub-domain is given to each processor. The solution and the right hand side vectors are partitioned in the same way. This distribution guarantees that communications take place only along the sub-domain boundaries.

Note on intersections

In the parallel implementation of the method the search for intersection is local, i.e. every processor looks for the intersections in its own sub-domain and it fills parallel vectors defined on the regular grid in with information about existence and position of the intersections. This guarantees the best performance but it needs communications to update the ghost regions. However, without a change in the arrangement every processor would number the intersections in an incompatible way compared with other processors. In order to avoid any ambiguity, every processor starts the search in its own domain, Figure 4.13a; then, a sequential vector owned by all the processors is updated: every position in this vector is associated to a processor and stores the number of unknowns (grid points plus intersections) in every sub-domain. The sum of the vector elements, which come before the position associated to a processor, is used to increase the intersections arrangement of that processor, and a global numeration of the intersections is obtained, Figure 4.13b. By this way we can get as less communications as we can in solving the linear system.



(a) Example of intersections search (4 processors case). The local arrangements; the sequential vector stores the values $(29, 29, 29, 29)$. Intersections in red.

(b) Example of intersections search (4 processors case). The final global arrangement. Intersections in red.

Figure 4.13: Intersections arrangement.

Aspects of the method simplifying parallelization through PETSc library

Some aspects of this method make the parallelization easy to be accomplished, comparing the present one to the already existing methods. The use of the Cartesian grid and the level set function, putting information about the intersections on the Cartesian grid points make easy to distribute the interface among the processors, just partitioning the computational domain. Moreover, the discretization of the transmission conditions is performed without introducing local reference systems or local sub-problems, such as matrix inversions ([27]), optimization problems ([59]) or further jump conditions calculations ([17] and [89]). Surely these methods can simplify the matrix of the linear system, but it may be hard to preserve a good balance in processors computational load.

Concerning the use of the PETSc library, the most important structure to manage Cartesian grids is the *DA*, the distributed array. Even if we have to deal with the intersection points, storing the information about them on the grid nodes makes the use of the *DA* helpful to manage these points. Although the augmented nature of the linear system seems to complicate the extraction of the solution on the grid nodes, efficient routines and the partitioning introduced make the scattering between parallel vectors easy to be implemented with few code lines. Furthermore, we have to implement this operation just for one time at the end of the code with almost no computational cost (compared to the one needed to solve the linear system). All these aspects makes parallel code quite similar to the sequential one, provided the right use of the PETSc structures and routines.

Petsc structures

Some of the PETSc data and logical structures are introduced in this section, providing some indications about their use in the parallel implementation of the

method.

- *Vec*, it is the PETSc structure for storing parallel vectors; it is the simplest structure PETSc provides. We used it to store the right hand side, the solution and to manage all the information about the intersections between the grid and the interface.

- *Mat*, it is the PETSc structure for storing parallel matrices; in PETSc you can choose different parallel implementation for the matrices. In this case the right format is the compressed sparse row one which is the default. Obviously we used it to store the matrix A

- *DA*, distributed array, it is to be used in conjunction with vectors and it allows the management of structured grids. It is a logical regular and rectangular structure and it is not intended to store data (the storage is accomplished by Vec and Mat), but specifically to contains information about the layout of the parallel data and about communications. Here the DA has been used not only for the assembly of the differential operator discretization matrix, but also to implicitly coordinate the communications needed to update the ghost regions in the data structures. It also contains the width of the ghost regions which is 3 for our purpose.

- *IS*, index set, it generally facilitates the communications in unstructured grids. Apparently it should be useless in a parallelization on a structured grid, but on the contrary the presence of new off-grid unknowns makes it crucial to extract the solution on the grid from the augmented vector u.

- *VecScatter*, it relates different index sets allowing scatters of data from one vector to one another. It is to be used with IS and, as this one, it is the base object to efficiently extract the solution on the regular grid.

- *KSP*, it is the heart of PETSc. It provides an efficient access to the parallel linear solvers. As an interface it allows the use of the combination of Krylov subspace methods and preconditioners for the iterative solution of linear systems. Once the matrix and the right hand side are assembled and the solution allocated, it needs just few calls to solve the system choosing the solver and the preconditioner at runtime.

- *PC*, it allows the management of the preconditioners. It is the part of the KSP interface for choosing the preconditioner and setting it up.

Petsc routines

A very schematic idea of the code is given here through an overview of the main PETSc routines involved in the parallel implementation of the code.

Once the intersections have been correctly arranged, no explicit calls to MPI routines is needed to correctly assemble parallel matrix and vectors with correct ghost regions, to solve the system and to extract the solution on the computational grid, provided that appropriate PETSc routines are used. Have the DA been defined through *DACreate()* routine, the parallel assembly of the matrix and the right hand side can be performed using *MatCreateMPI()*, *VecCreateMPI()*, *MatSetValues()*, *VecSetValues()* and the relative assembly routines.

The DA allows to use the correct numeration of the grid points which is usually different from the application one. Such a difference can be experienced in [13] and in Figure 4.13.

When the elements of the linear system are ready to be used, the routines *KSPCreate()*, *KSPSetOperators()*, *KSPSetUp()*, *KSPSolve()* are sufficient to pass the matrix A, the RHS f to the solver and to get the solution u. The iterative solver can be chosen at runtime provided that one calls *KSPSetFromOptions()*. For Neumann boundary conditions a *NullSpace* object has to be created and associated to the *KSP*, through *MatNullSpaceCreate()* and *KSPSetNullSpace()*.

Because of the augmented nature of the solution, a procedure to extract the solution on the regular grid is needed and it is achieved by *ISCreateGeneral()*, *VecScatterCreate()*, *VecScatterBegin()*, *VecScatterEnd()*.

Providing the details is out of the aim of this text, but it is remarkable that some routines operates on global vectors, while others works on local vectors (local and global are to be intended in the sense of the domain of a single processor or of the whole computational domain). This makes the use of routines updating the ghost regions crucial as well as straightforward, working with the *DA* structure: *DALocalToGlobal()*, *DAGlobalToLocal()* and *DALocalToLocal()* are all we need to consider any kind of ghost region updating.

4.6 Numerical validation of the method

4.6.1 Sequential validation of the method in two dimensions

In this section we present the convergence results for several test cases in two dimensions. Some of them have already been studied by others authors, which allow us to compare our results to theirs, in terms of convergence order as well as error amplitude. The linear systems are solved with routines from the SPARSKIT library [80]: a GMRES algorithm with a ILU preconditionning. In all the following test cases, we consider a square domain Ω consisting in the union of two subdomains Ω_1 and Ω_2 separated by an interface Σ. If not specified otherwise, $\Omega = [-1, 1] \times [-1, 1]$. We impose exact Dirichlet boundary conditions on the outer boundary of Ω.

Problem 1

The equations (4.1 - 4.3) are solved imposing the following exact solution in $\Omega = [-2, 2] \times [-2, 2]$, with $r = \sqrt{x^2 + y^2}$:

$$u(x,y) = \begin{cases} 100 & \text{if } r = 1 \\ 100 + 50\ ln(1/r) & \text{if } 1 < r < 1.5 \\ 100 + 50\frac{k_1}{k_2} ln(1/r) + 50\ (1 - \frac{k_1}{k_2})\ ln(1/1.5) & \text{if } r > 1.5 \end{cases} \quad (4.43)$$

where $k = k_1$ for $1 < r < 1.5$, $k = k_2$ for $r > 1.5$.

Numerical results for $k_1 = 2$ or 1000 and $k_2 = 1$ for the discrete L^∞ norm are presented on Table 4.1 and Table 4.3. On Table 4.2 the results without stabilization for $k_1 = 2$ are presented. Plots of the numerical solution and

the numerical error for $n_x = n_y = 120$ are presented in Figures 4.14 and 4.15 for $k_1 = 2$ and in Figures 4.16 and 4.17 for $k_1 = 1000$. We observe global second order numerical convergence if the stabilization is applied. If not, the convergence order oscillates, and the accuracy is deteriorated for some values of the grid points number.

In the case of $k_1 = 1000$ the error in maximum norm may seem very big for the low resolutions. But the maximum norm of the solution itself depends also of the value of k_1. If the error were expressed in a normalized maximum norm error, the contrast with the error obtained in the case $k_1 = 2$ would be attenuated.

N	L^1 error	order	L^∞ error	order
30	4.390×10^{-1}	-	8.590×10^{-2}	-
60	1.364×10^{-1}	1.69	3.474×10^{-2}	1.31
120	2.581×10^{-2}	2.04	6.351×10^{-3}	1.88
240	7.580×10^{-3}	1.95	1.653×10^{-3}	1.90
360	3.079×10^{-3}	1.99	7.435×10^{-4}	1.91
480	2.333×10^{-3}	1.89	4.269×10^{-4}	1.91

Table 4.1: Numericals results for Problem 1, for $k_1 = 2$ and $k_2 = 1$.

N	L^1 error	order	L^∞ error	order
30	2.735×10^{-1}	-	6.651×10^{-2}	-
60	5.528×10^{-2}	2.31	3.386×10^{-2}	0.97
120	1.648×10^{-1}	0.36	5.784×10^{-2}	0.20
240	5.915×10^{-3}	1.84	1.638×10^{-3}	1.78
360	4.915×10^{-3}	1.62	1.905×10^{-3}	1.43
480	1.279×10^{-3}	1.94	4.287×10^{-4}	1.82

Table 4.2: Numericals results for Problem 1, for $k_1 = 2$ and $k_2 = 1$, without stabilization.

N	L^1 error	order	L^∞ error	order
30	56.567×10^{0}	-	19.147×10^{0}	-
60	13.502×10^{0}	2.07	5.110×10^{0}	1.91
120	3.128×10^{0}	2.09	1.459×10^{0}	1.86
240	7.638×10^{-1}	2.07	3.436×10^{-1}	1.93
480	2.009×10^{-1}	2.04	9.651×10^{-2}	1.91

Table 4.3: Numericals results for Problem 1, for $k_1 = 1000$ and $k_2 = 1$.

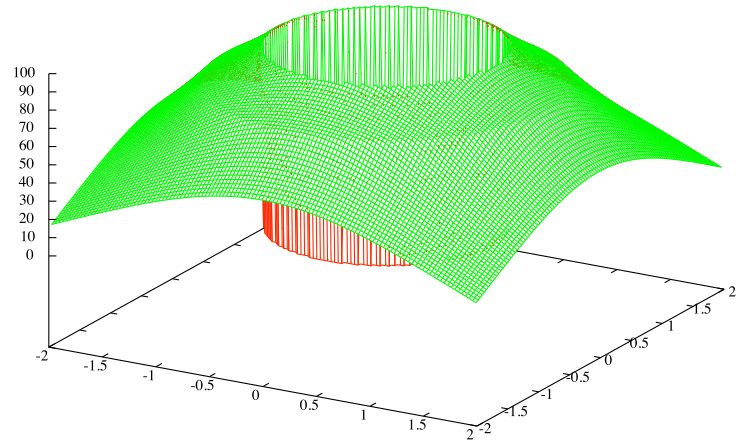


Figure 4.14: Numerical solution for $n_x = n_y = 120$ for Problem 1 with $k_1 = 2$ ($u = 0$ for $r < 1$)

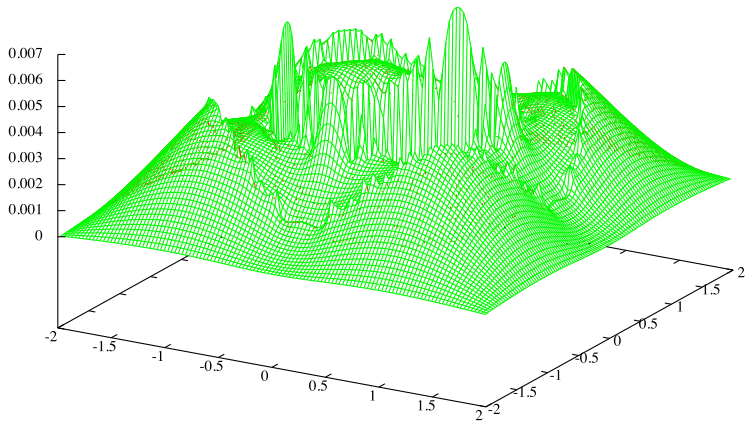


Figure 4.15: Numerical error for $n_x = n_y = 120$ for Problem 1 with $k_1 = 2$

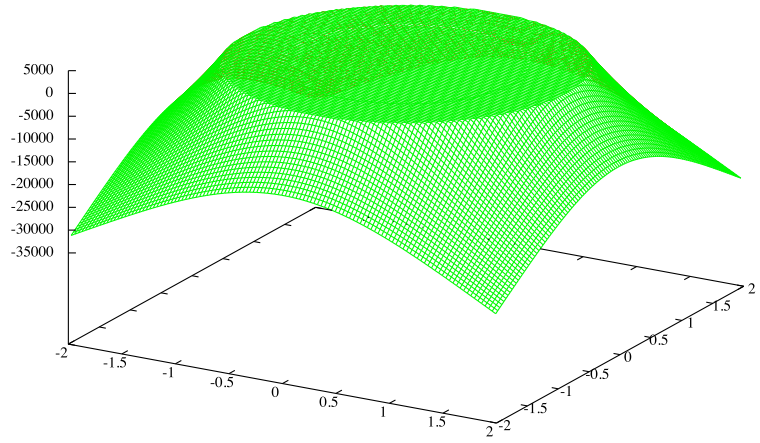


Figure 4.16: Numerical solution for $n_x = n_y = 120$ for Problem 1 with $k_1 = 1000$ ($u = 0$ for $r < 1$)

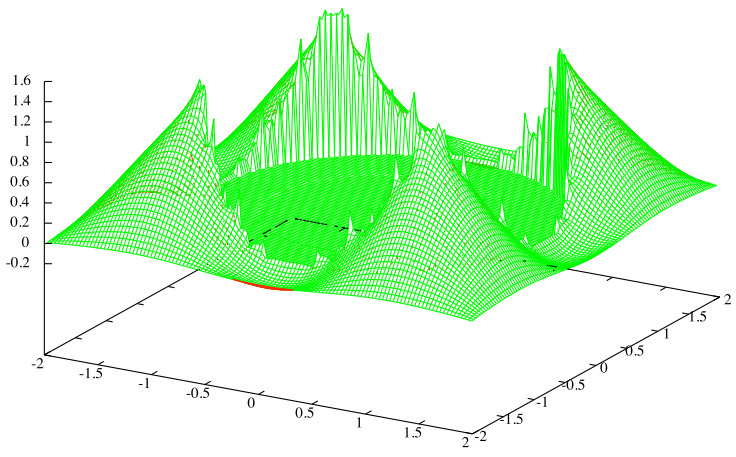


Figure 4.17: Numerical error for $n_x = n_y = 120$ for Problem 1 with $k_1 = 1000$

Problem 2

We compute now the solution for a more complex interface in $\Omega = [-2, 2] \times [-2, 2]$. Ω_1 is defined by the intersections of the circles of radius 1.5 whose centres are located on points $(1.6, 1.6)$, $(1.6, -1.6)$, $(-1.6, 1.6)$ and $(-1.6, -1.6)$. Ω_2 consists in $\Omega \setminus \Omega_1$. Σ is the interface between Ω_1 and Ω_2.

The exact solution is:

$$u(x, y) = \begin{cases} cos(x) + cos(y) \text{ in } \Omega_1 \\ cos(y)e^x \text{ in } \Omega_2. \end{cases} \tag{4.44}$$

Numerical results for $k_1 = 2$ or 100 in Ω_1 and $k_2 = 1$ in Ω_2 for the discrete L^∞ norm are presented on Table 4.4. Plots of the numerical solution and the numerical error for $n_x = n_y = 80$ are presented on Figures 4.18 and 4.19. This test-case shows that our method maintains its second order accuracy in the case of a complex interface with sharp edges.

N	L^∞ error	order
20	7.191×10^{-3}	-
40	7.799×10^{-4}	3.22
80	2.334×10^{-4}	2.47
160	6.852×10^{-5}	2.24
240	2.886×10^{-5}	2.22
320	1.711×10^{-5}	2.18
400	1.057×10^{-5}	2.18

Table 4.4: Numericals results for Problem 2

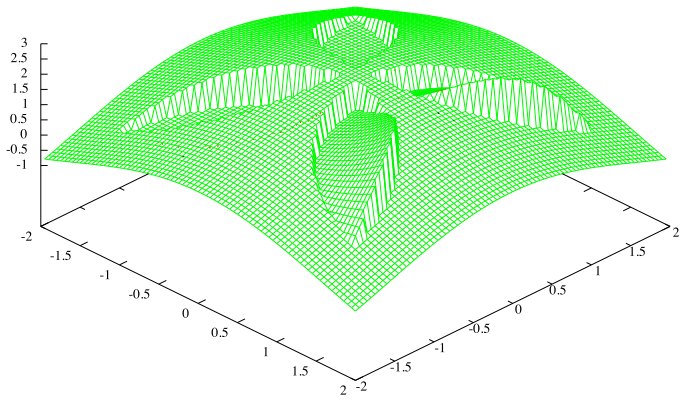


Figure 4.18: Numerical solution for $n_x = n_y = 80$ for Problem 2

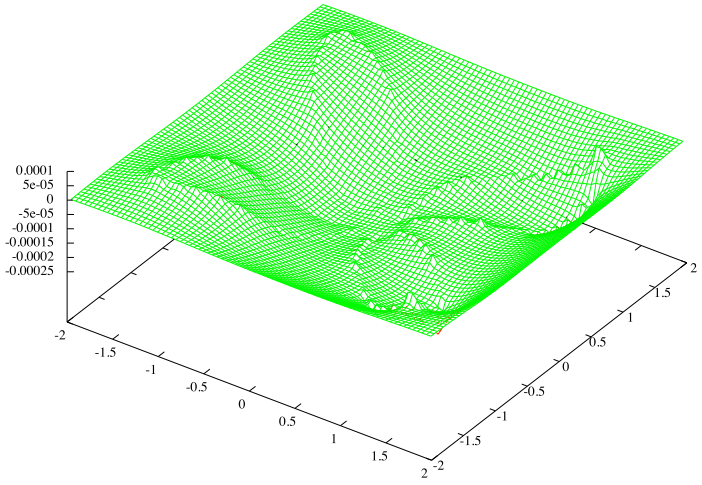


Figure 4.19: Numerical error for $n_x = n_y = 80$ for Problem 2

Problem 3

It is a test case appearing in [95] (MIB method, case 3 of the tests on irregular interfaces) and [27] (CIM, example 4). We consider an elliptical interface Σ defined as:

$$(\frac{x}{18/27})^2 + (\frac{y}{10/27})^2 = 1. \tag{4.45}$$

The exact solution is:

$$u(x,y) = \begin{cases} e^x cos(y), \text{ inside } \Sigma \\ 5e^{-x^2 - \frac{y^2}{2}}, \text{ otherwise.} \end{cases} \tag{4.46}$$

As in [95] we fix the diffusion coefficient k to be 1 outside the interface, and we choose 10 or 1000 inside the interface. For this test case we still observe the second order convergence, as presented in Tables 4.5 and 4.6. The numerical solution and error are plotted in Figures 4.20 and 4.21 for $k = 10$ inside Σ and in Figures 4.22 and 4.23 for $k = 1000$ inside Σ. For $k = 1000$ our method provides smaller errors than the two others Cartesian methods. For $k = 10$ we obtain more accurate results than with the MIB method, and slightly less accurate than with the CIM. In the case $k = 1000$ inside the interface, we also present in Table 4.6 the numerical results obtained with the 8 points version of the method. We see that the 8 points version provides less accurate results than the 13 points version. The computational time is with the 13 points version 0.051 s for 80^2 grid points, 0.191s for 160^2 grid points and 0.763 s for 320^2 grid points, while it is 0.049 s for 80^2 grid points, 0.189s for 160^2 grid points and 0.769 s for 320^2 grid points with the 8 points version.

N	L^∞ error	order	L^∞ error for MIB [95]	L^∞ error for CIM [27]
20	8.115×10^{-3}	-	2.659×10^{-2}	4.067×10^{-3}
40	9.152×10^{-4}	3.15	5.206×10^{-3}	6.171×10^{-4}
80	3.221×10^{-4}	2.33	1.487×10^{-3}	1.682×10^{-4}
160	6.335×10^{-5}	2.33	3.746×10^{-4}	3.975×10^{-5}
320	1.212×10^{-5}	2.35	7.803×10^{-5}	7.390×10^{-6}

Table 4.5: Numericals results for Problem 3, for $k = 10$ inside the interface

N	L^∞ error (13 points)	order	L^∞ error (8 points)	L^∞ error for MIB [95]	L^∞ error for CIM [27]
20	1.083×10^{-1}	-	1.005×10^{-1}	9.130×10^{-2}	3.539×10^{-1}
40	4.094×10^{-2}	1.40	4.715×10^{-2}	2.764×10^{-2}	1.100×10^{-1}
80	7.045×10^{-3}	1.97	2.967×10^{-2}	7.524×10^{-3}	2.028×10^{-2}
160	1.824×10^{-3}	1.96	7.979×10^{-3}	2.169×10^{-3}	6.462×10^{-3}
320	4.671×10^{-4}	1.97	2.120×10^{-3}	4.841×10^{-4}	1.437×10^{-3}

Table 4.6: Numericals results for Problem 3, for $k = 1000$ inside the interface

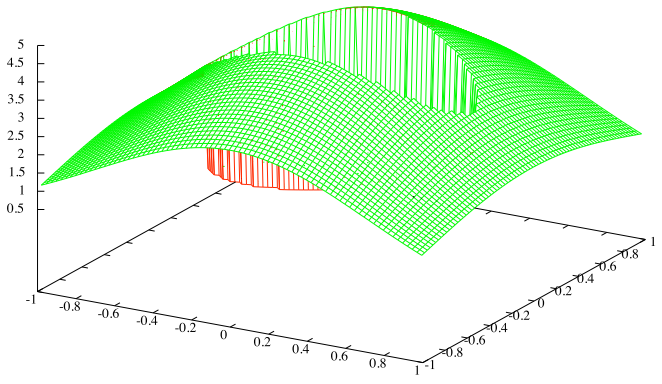


Figure 4.20: Numerical solution for $n_x = n_y = 80$ for Problem 3 with $k = 10$ inside the interface

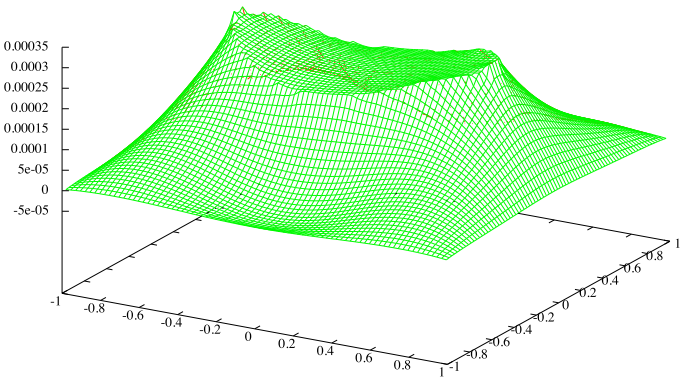


Figure 4.21: Numerical error for $n_x = n_y = 80$ for Problem 3 with $k = 10$ inside the interface

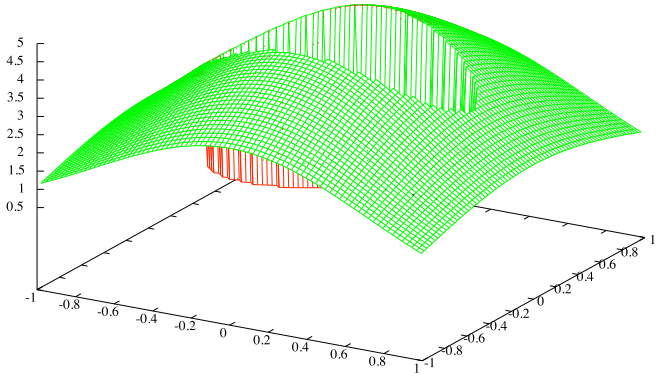


Figure 4.22: Numerical solution for $n_x = n_y = 80$ for Problem 3 with $k = 1000$ inside the interface

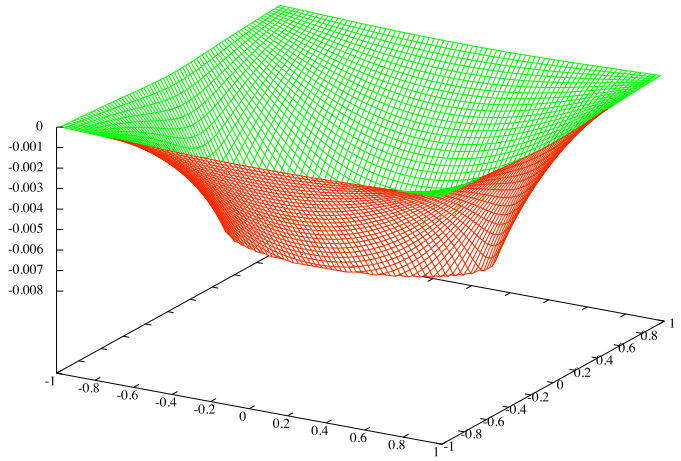


Figure 4.23: Numerical error for $n_x = n_y = 80$ for Problem 3 with $k = 1000$ inside the interface

Problem 4

It is a test case studied in several references: [57], [17], [89], [59] and [27]. A slightly different problem is also considered in [93]. We consider an circular interface Σ defined by:

$$r^2 = 1/4 \tag{4.47}$$

with $r = \sqrt{x^2 + y^2}$. The coefficient k varies in space:

$$k(x, y) = \begin{cases} r^2 + 1 \text{ inside } \Sigma \\ b \text{ otherwise.} \end{cases} \tag{4.48}$$

The exact solution is:

$$u(x, y) = \begin{cases} r^2 \text{ inside } \Sigma \\ (1 - \frac{1}{8b} - \frac{1}{b})/4 + (\frac{r^4}{2} + r^2)/b + C\, log(2\, r)/b \text{ otherwise,} \end{cases} \tag{4.49}$$

with b a parameter appearing in the formula for the coefficient k and for the solution u, that we make vary: $b = 10, 1000$ and 0.001. The source term is :

$$f(x, y) = 8(x^2 + y^2) + 4. \tag{4.50}$$

The numerical results and orders of convergence are presented in Tables 4.7, 4.8 and 4.9. The numerical solution and error are plotted in Figures 4.24,4.25, 4.26,4.27, 4.28 and 4.29. For this test we again observe second order accuracy. If we compare to the results of other Cartesian methods found in the literature, the current method is globally one of the most accurate.

N	Current method	IIM [57]	DIIM [17]	EJIIM [89]	MIIM [59]	CIM[27]
20	4.623×10^{-4}	3.5195×10^{-3}	5.378×10^{-4}	7.6×10^{-4}	-	1.259×10^{-3}
40	1.364×10^{-4}	7.5613×10^{-4}	1.378×10^{-4}	2.4×10^{-4}	4.864×10^{-4}	2.565×10^{-4}
80	4.431×10^{-5}	1.6512×10^{-4}	3.470×10^{-5}	7.9×10^{-5}	1.448×10^{-4}	5.215×10^{-5}
160	1.568×10^{-5}	3.6002×10^{-5}	8.704×10^{-6}	2.2×10^{-5}	3.012×10^{-5}	1.142×10^{-5}
320	7.053×10^{-6}	8.4405×10^{-6}	2.177×10^{-6}	5.3×10^{-6}	8.226×10^{-6}	2.725×10^{-6}

Table 4.7: Numericals results in L^∞ norm for Problem 4, $b = 10$.

N	Current method	Order	DIIM [17]	MIIM [59]	CIM [27]
32	1.825×10^{-4}	-	2.083×10^{-4}	5.136×10^{-4}	2.732×10^{-4}
64	4.965×10^{-5}	1.88	5.296×10^{-5}	8.235×10^{-5}	3.875×10^{-5}
128	1.304×10^{-5}	1.90	1.330×10^{-5}	1.869×10^{-5}	5.337×10^{-6}
256	3.333×10^{-6}	1.92	3.330×10^{-6}	4.026×10^{-6}	7.241×10^{-7}

Table 4.8: Numericals results in L^∞ norm for Problem 4, $b = 1000$.

N	Current method	Order	DIIM [17]	MIIM [59]	CIM [27]
32	2.036×10^{0}	-	4.971×10^{0}	9.346×10^{0}	4.278×10^{-1}
64	3.522×10^{-1}	2.53	1.176×10^{0}	2.006×10^{0}	1.260×10^{-1}
128	7.255×10^{-2}	2.41	2.900×10^{-1}	5.808×10^{-1}	3.773×10^{-2}
256	1.807×10^{-2}	2.27	7.086×10^{-2}	1.374×10^{-1}	1.365×10^{-2}

Table 4.9: Numericals results in L^∞ norm for Problem 4, $b = 0.001$.

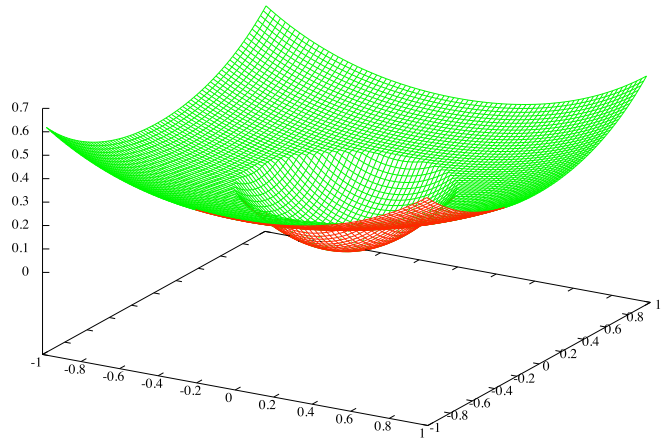


Figure 4.24: Numerical solution for $n_x = n_y = 80$ for Problem 4 with $b = 10$

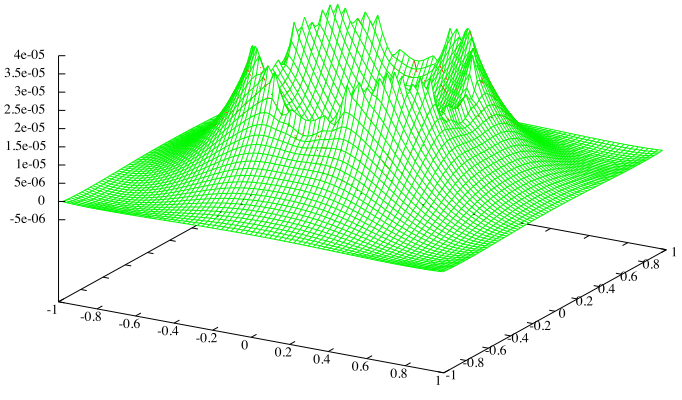


Figure 4.25: Numerical error for $n_x = n_y = 80$ for Problem 4 with $b = 10$

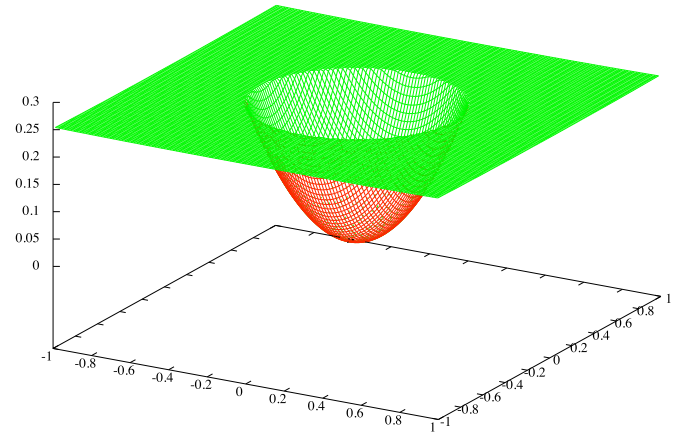


Figure 4.26: Numerical solution for $n_x = n_y = 80$ for Problem 4 with $b = 1000$

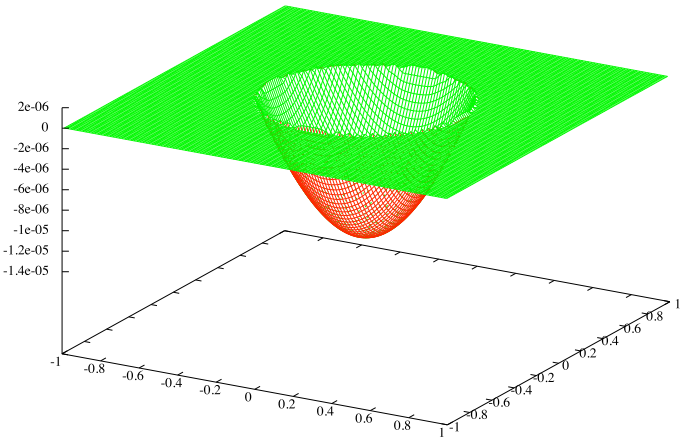


Figure 4.27: Numerical error for $n_x = n_y = 80$ for Problem 4 with $b = 1000$

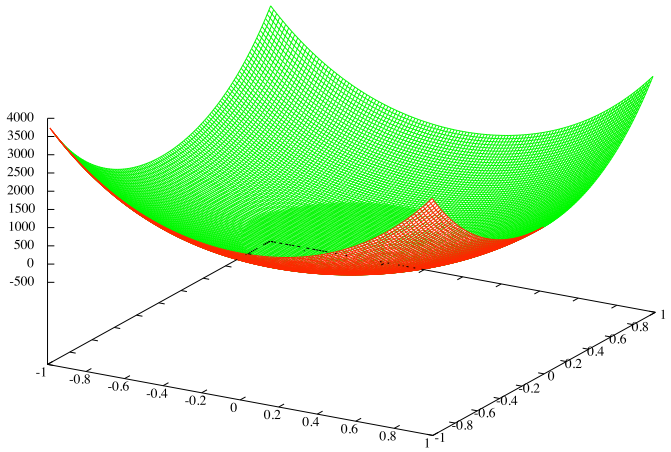


Figure 4.28: Numerical solution for $n_x = n_y = 80$ for Problem 4 with $b = 0.001$

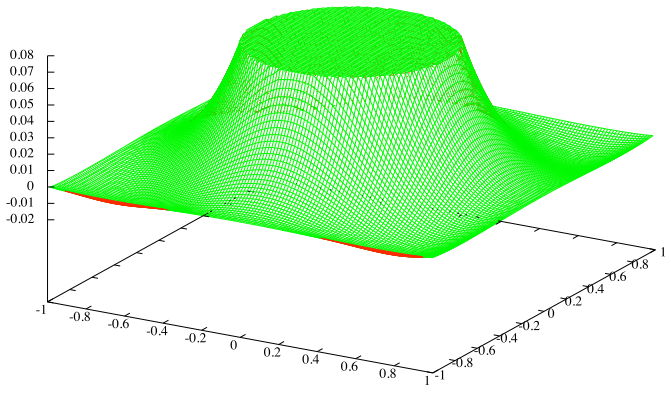


Figure 4.29: Numerical error for $n_x = n_y = 80$ for Problem 4 with $b = 0.001$

4.6.2 Numerical study of the parallel method in two dimensions

In order to describe the performances of the parallel code implementing the method, we conduct here some numerical experiments on a fixed grid ($N_x = 3500, N_y = 3500$), varying the number of processors, and on finer and finer grids to study the error convergence rate.

Problem 1

We study the Problem 1, presented in section 4.6.1, with $k_1 = 1000$ and $k_2 = 1$. For our tests we chose the restarted GMRES solver and the Additive Schwarz Method (ASM) preconditioner with overlapping between matrix blocks assigning one block per process and using the ILU preconditioner on each block. The scalability results are presented in Figure 4.30 and in Table 4.11. It shows the relationship between the number of processors and the calculation time. The data have been fitted with a power law, $t = aN^b$, where t is the calculation time, N is the number of processors and an estimation of the parameters a and b is given. The trend of the data implies we are not too far from a perfect parallelism, $b = -1$. The error convergence rate results are presented in Table 4.10. A global second order numerical convergence is observed for the parallel implementation of the method too.

N	L^1 error	order	L^∞ error	order
800	3.381×10^{-2}	-	1.468×10^{-1}	-
1600	9.074×10^{-3}	1.90	3.960×10^{-2}	1.89
3200	2.346×10^{-3}	1.92	1.020×10^{-2}	1.92
6400	6.480×10^{-4}	1.91	2.698×10^{-3}	1.92

Table 4.10: Parallel numericals results for Problem 1, for $k_1 = 1000$ and $k_2 = 1$.

Number of processors	time (sec)
4	344.25
9	158.22
16	100.60
25	71.54
32	62.40
49	44.26

Table 4.11: Scalability results on a 3500×3500 grid for Problem 1.

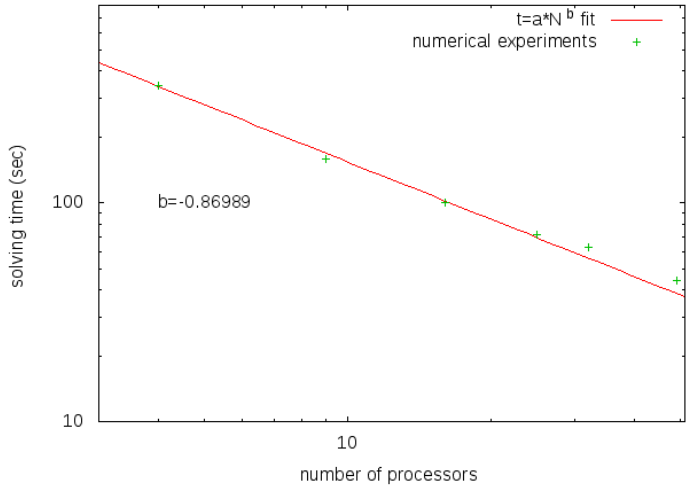


Figure 4.30: This figure shows how the calculation time scales with the number of processors. Crosses: experimental data in Table 4.11. Line: least square fit of the data. The experiments have been conducted on the machine Fourmi at PlaFRIM (see, [3])

Problem 5

It is a test case mentioned in [58] and [27]. We consider a flower-like interface Σ:

$$\phi(r,\theta) = r - r_0 - 0.2\sin(\omega\theta). \tag{4.51}$$

with $r = \sqrt{(x-x_c)^2+(y-y_c)^2}$, $\theta = arctan((y-y_c)/(x-x_c))$, $x_c = y_c = 0.2\sqrt{20}$. The coefficient k is defined by:

$$k(x,y) = \begin{cases} k^- \text{ inside } \Sigma \\ 1 \text{ otherwise.} \end{cases} \tag{4.52}$$

The exact solution is:

$$u(x,y) = \begin{cases} e^x cos(y), \text{ inside } \Sigma \\ 5e^{-x^2-\frac{y^2}{2}}, \text{ otherwise.} \end{cases} \tag{4.53}$$

We study two cases: $\omega = 5$, $r_0 = 0.5$, $k^- = 1000$, and $\omega = 12$ and $r_0 = 0.4$, $k^- = 100$. In Figure 4.31 for $\omega = 5$ and in Figure 4.32 for $\omega = 12$ the numerical results are presented for N varying from 270 to 3500, with an increment 10 from 270 to 1090, as in [58] and [27], and an increment 100 from 1100 to 3500. The numerical solution and error are plotted in Figures 4.33, 4.34,4.35 and 4.36. We observe in both cases a second order convergence. More precisely, in the case $\omega = 5$ the linear regression slope is 2.06, and in the case $\omega = 12$ it is 2.64, but the convergence rate is amplified by the presence of oscillations for the smallest values of N. The overall accuracy is slightly better than CIM [27] for $\omega = 5$ and slightly worse for $\omega = 12$, the latter being itself globally more accurate than in

[58]. The appearance of the plots in Figures (4.31)-(4.32) is similar to the one of the plots for the same tests in [27]. The oscillations are mainly due to the position of the interface relative to the grid and to the consequent presence of order one approximation of the co-normal jump condition (see Figure 4.11). In [27] the authors consider these larger errors due to the pollution from this order one scheme, calling the points where this scheme has to be applied exceptional points.

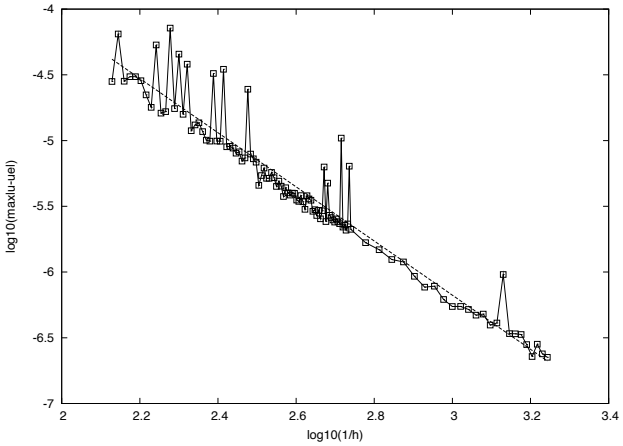


Figure 4.31: Convergence test for Problem 5 with $\omega = 5$, $r_0 = 0.5$ and $k^- = 1000$. Dashed line illustrates the slope of order two accuracy. Solid line is the slope of the linear regression.

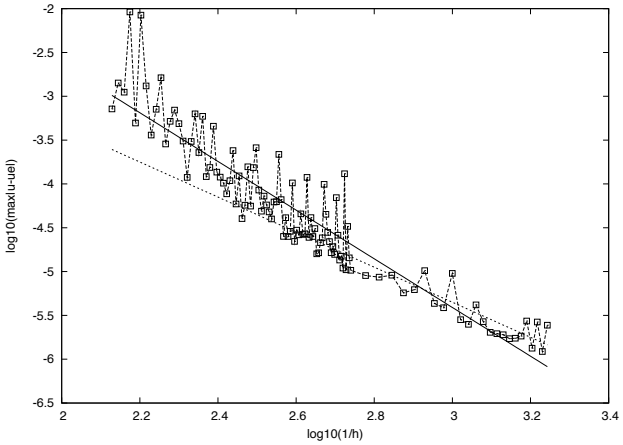


Figure 4.32: Convergence test for Problem 5 with $\omega = 12$, $r_0 = 0.4$ and $k^- = 100$. Dashed line illustrates the slope of order two accuracy. Solid line is the slope of the linear regression.

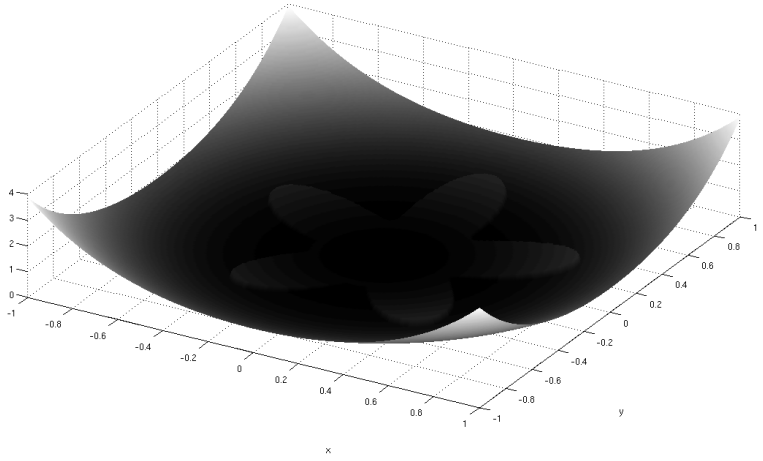


Figure 4.33: Numerical solution for $n_x = n_y = 270$ for Problem 5 for $\omega = 5$.

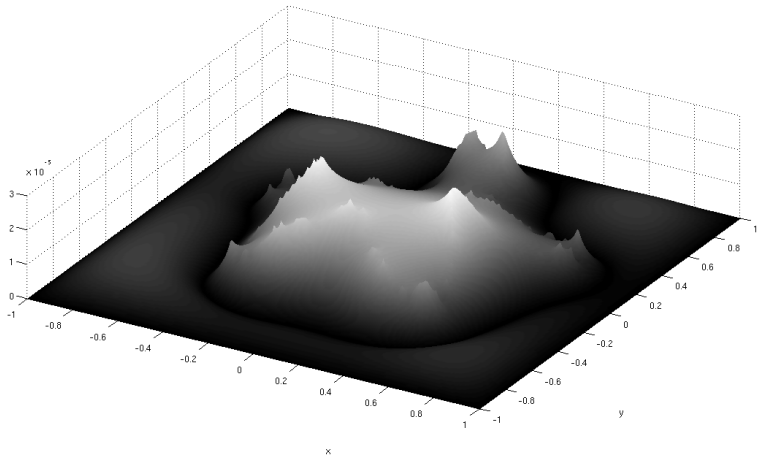


Figure 4.34: Numerical error for $n_x = n_y = 270$ for Problem 5 for $\omega = 5$.

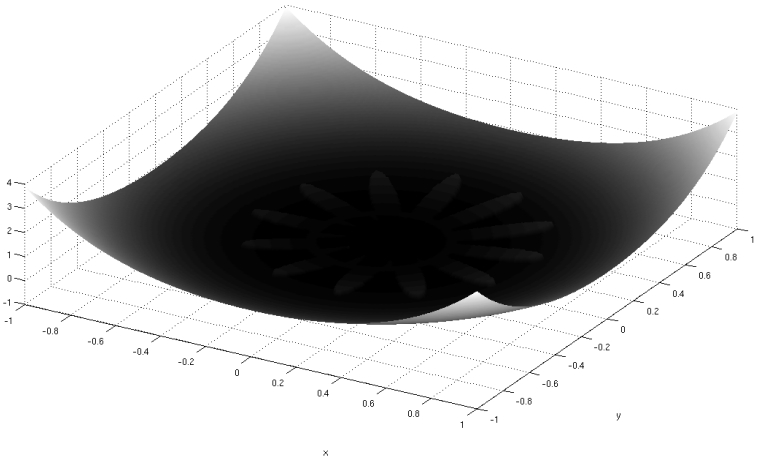


Figure 4.35: Numerical solution for $n_x = n_y = 270$ for Problem 5 for $\omega = 12$.

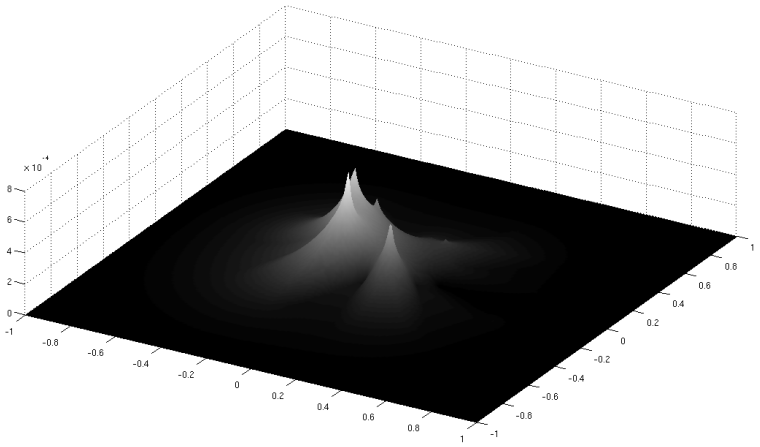


Figure 4.36: Numerical error for $n_x = n_y = 270$ for Problem 5 for $\omega = 12$.

4.6.3 Numerical study of the parallel method in three dimensions. A simple case.

In this section we introduce the first case we have studied in order to start the validation of the method for three-dimensional problems. The dimension-by-dimension approach used in building the method makes us to think that what we will see in three dimensions won't be so different from what we have seen in two dimensions. On the other hand, further test cases with more complex interfaces are surely necessary to corroborate this hypothesis. Error convergence analysis and scalability performance test are provided for the following problem.

The equations (4.1)-(4.3) are solved in the domain $\Omega = [-5,5] \times [-5,5] \times [-5,5]$ with the analytical solution

$$u(x,y,z) = \begin{cases} 100 & \text{if } r = 1 \\ 50 + \frac{50}{r} & \text{if } 1 < r \leq 3 \\ 50\frac{k_2}{k_1}\frac{1}{r} + (1 - \frac{k_2}{k_1})\frac{50}{3} + 50, & \text{otherwise} \end{cases} \tag{4.54}$$

with $k = k_1$ for $1 < r \leq 3$ and $k = k_2$ for $r > 3$. Tests have been conducted for $k_1 = 1000, k_2 = 1$ and for $k_1 = 1, k_2 = 1000$. Similarities with Problem 1 in Section 4.6 are evident.

Error convergence results for the two cases are in Tables 4.12 and 4.13. An example of numerical solution and error are provided by Figures 4.37 and 4.38. We can say that, at least for this single case, the method has a good second order accuracy both in L^∞-norm and in L^2-norm. For this geometry, as for the analogous two-dimensional geometry in Problem 1, there are no intersections with first order approximation of equation 4.3. This is the reason for which this test case can be just preliminary and more complex interfaces need to be tested in three dimensions. However, on the other side, this test is good enough to validate the method for cases with pure second order approximation of the co-normal jump condition.

N	L^∞ error	order	L^2 error	order
40	4.50×10^{-2}	-	3.29×10^{-3}	-
80	1.05×10^{-2}	2.09	7.37×10^{-4}	2.16
160	2.49×10^{-3}	2.07	1.73×10^{-4}	2.09
320	6.04×10^{-4}	2.04	4.20×10^{-5}	2.04

Table 4.12: Error convergence results for the three-dimensional test case with $k_1 = 1000$ and $k_2 = 1$.

We also conduct some numerical experiments on a fixed grid to explore the scalability features of the method for three-dimensional problems. We choose the largest grid considered in the error convergence tests, i.e $N_x = N_y = N_z = 320$. The results are shown in Figure 4.39. The solving time scales with the number of processors with a good exponent. Moreover, the larger the processors get, the better this exponent seems to get.

N	L^∞ error	order	L^2 error	order
40	3.798×10^{0}	-	1.103×10^{0}	-
80	1.006×10^{0}	1.91	2.849×10^{-1}	1.95
160	2.593×10^{-1}	1.95	7.239×10^{-2}	1.98
320	6.603×10^{-2}	1.97	1.820×10^{-2}	1.99

Table 4.13: Error convergence results for the three-dimensional test case with $k_1 = 1$ and $k_2 = 1000$.

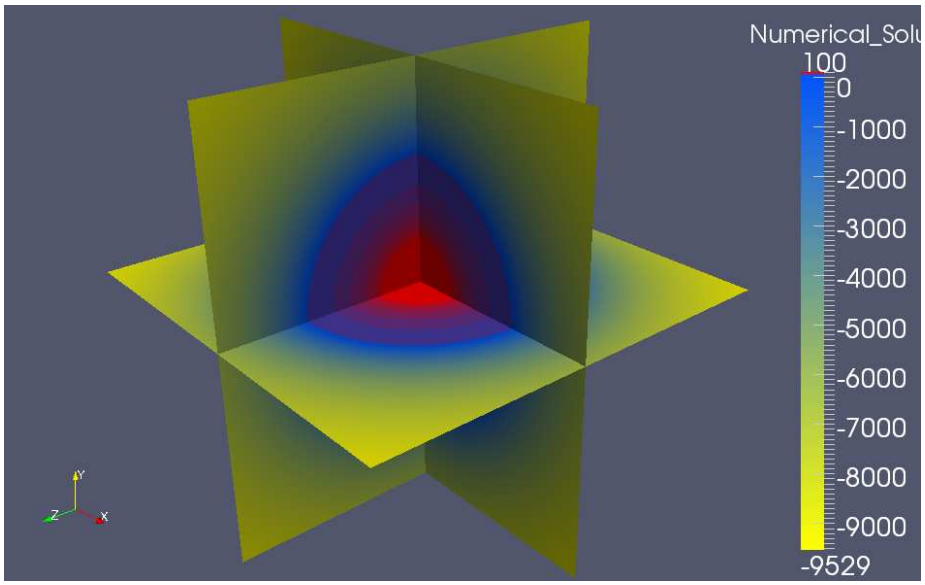


Figure 4.37: Numerical solution for the 3D test case. $Nx = N_y = N_z = 160$, $k_1 = 1$ and $k_2 = 1000$

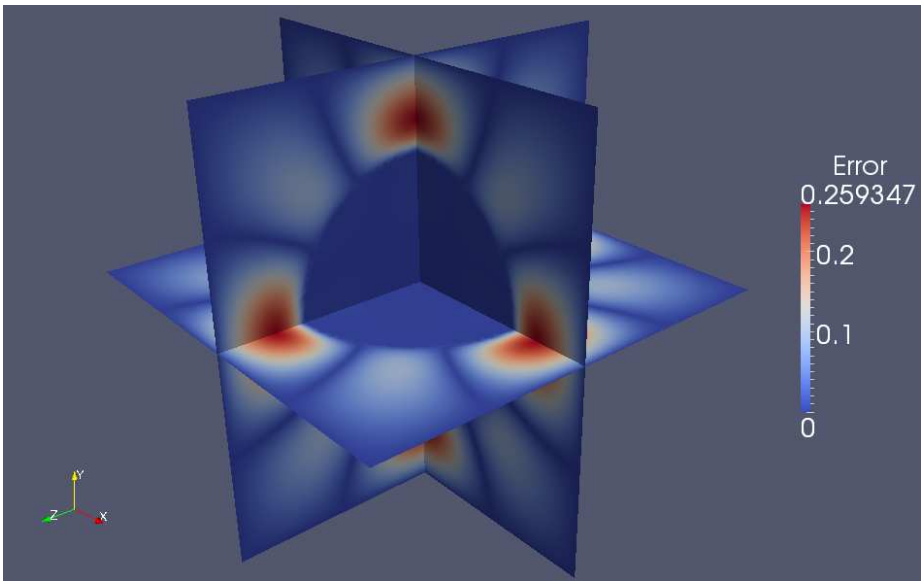


Figure 4.38: Numerical error for the 3D test case. $Nx = N_y = N_z = 160$, $k_1 = 1$ and $k_2 = 1000$

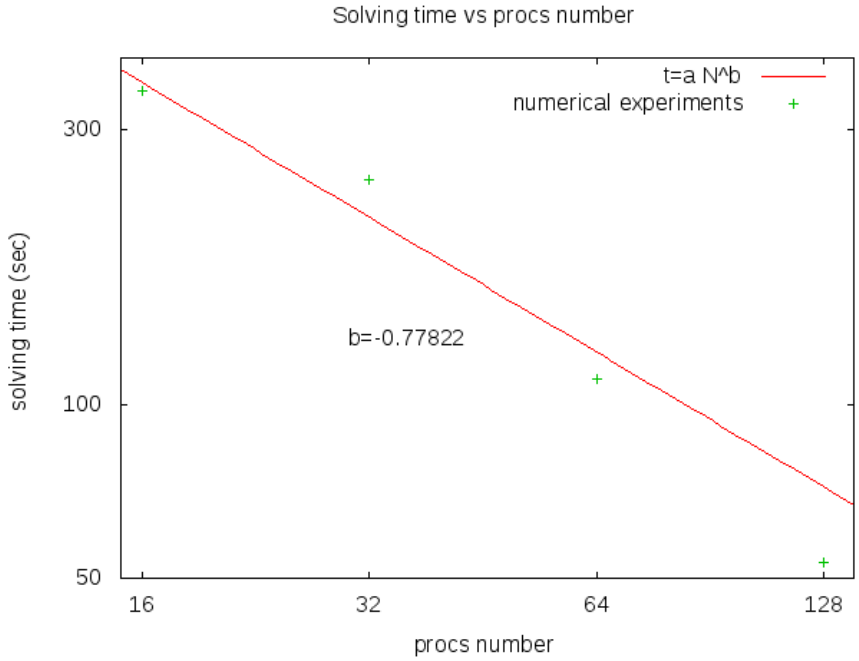


Figure 4.39: Scalability results on a $320 \times 320 \times 320$ grid for $k_1 = 1000$ and $k_2 = 1$.

4.7 Conclusions about the method

In this chapter we have presented a parallel second order Cartesian method to solve elliptic problems with discontinuous coefficients on interfaces. The method is based on a dimensional splitting and on the use of new unknowns located on the interface. The discretization of the elliptic operator near the interface is performed with standard finite differences formulas involving these interface unknowns, which are themselves determined by solving discretized flux transmissions equations at the interface. The use of interface unknowns allows to decouple the discretization of the elliptic operator from the treatment of the interface transmission conditions. This decoupling makes our method particularly simple to implement, and allow easy modifications in the discretization of the elliptic operator or of the interface jump conditions. Moreover, if the interface transmission conditions themselves are modified, only their discretization needs to be changed to solve the new problem. The method is parallelized with the PETSc library in a straightforward manner. It is second order accurate, even on complex interfaces, with an absolute error competitive with the others method of the literature, and its parallel implementation shows good scalability properties. The numerical discretization is based on a dimensional splitting. Considering one more dimension only requires to discretize the first and second derivatives in the new dimension with the same principles used for the other ones. A three-dimensional numerical experiment has been conducted with a simple spherical interface, showing good second order accuracy properties and good scaling performances. However, it needs to be tried in more stressing geometries, but we believe that our method is a good candidate to solve three-dimensional problems.

Chapter 5

Application of the method to the tumor growth modelling

In this chapter the introduced method is applied to solve elliptical irregular domain problems in the framework of the tumor growth modelling. This application is fully three-dimensional and it leads to the simulation of the evolution of a tumor nodule in a realistic domain, specifically a lung boundary.

5.1 Introduction to the tumor growth modelling

5.1.1 Brief notes on cancer biology

Before introducing the mathematical models of tumor growth, we want to outline some brief information about the biological nature of a cancer and its development, for the exclusive purpose of giving a reasonable idea of the chosen model. Far from being exhaustive, the following contains just the main ideas about such a complex disease, those ideas allowing the introduction of models and comparisons among them, being the latter able to lead to a choice.

Right from the beginning, it is better to specify that a *neoplasm* is the result of an abnormal proliferation of cells, i.e. a neoplasia, and if it has formed a lump, it is called *tumor*. Cancer is a malignant tumor, able to colonize nearby tissues and to spread to other parts of the body in a process called *metastasis*.

Fundamentally, cancer is the result of a failure to regulate those cellular functions determining the natural evolution of a cellular phenotype. In a neoplasm cell division, differentiation, programmed cell death (apoptosis) and genetic organization are inappropriately regulated. A correct control in these functions determines the equilibrium of a cellular population. Therefore, introducing how normal, natural human cells proliferate, differentiate, died and express their specific functions may help to comprehend what happens in cancer cells.

The proliferation is the increase in the number of cells as a results of cell growth and division. The proliferative potential of natural human cells, for sake of biologic evolution, exceeds the requirements of normal growth and development, but in healthy populations this great potential is controlled and limited during the cell cycle. A sketch of the cell cycle is given in Figure 5.1. In phase G1 the cells prepare the material (protein and RNA synthesis are active) for

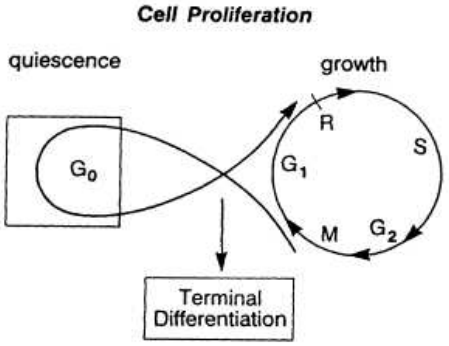


Figure 5.1: The cell cycle. From [8]

the successive functional phase S. After this preparatory phase cells can exit the cell cycle and enter in a G0 state, or *quiescent*, temporally or permanently, depending on extra-cellular conditions and cell maturation, or start the S phase. However, the transition from G0/G1 to S is highly regulated by the presence of important growth factors in R, the competence and restriction phase. In phase S the duplication of the genetic material takes place. The successive phase G2 has another checkpoint: the cells increase their size and, if no error in duplicated DNA is found, the cells are ready for the M phase. The mitosis process concludes the cycle in this phase whose result is the creation of two daughter cells. (See [8]).

The cell differentiation begins after the division and determines differences in phenotype. It depends on the gene expression, but the mechanisms regulating such expression are not completely understood, also because of its reliance on a wide variety of factors. Despite that, the most important aspect we must consider is the intimate connection of the proliferative ability of a cell and its state of differentiation, [8]. Immature, not yet differentiated cells, the stem cells, proliferate continuously, therefore their presence has to be carefully controlled to preserve the integrity and the equilibrium of a tissue.

The homeostasis of all tissues cannot rely only on a "production" process, the programmed cell death, apoptosis, trades the proliferation off. High regulation is needed for this process too, see [8] for details. This machinery is very important to control the cell survival, for example, correcting errors in cell cycle and avoiding neoplastic cell expansion.

Respecting the checkpoints in the cell cycle, completing the cell maturation or respecting the natural cell survival are delicate aspects and disrupting changes in their equilibria are the causes of cancer. It is commonly accepted that cancer origins from the effects of genetic mutations on the mechanisms controlling the cellular functions (see [8] for details on the *monoclonal* hypothesis), giving a certain group of cells the capability of breaking the constraints in proliferation, distorting the differentiation program, creating abnormal variant cells or avoiding programmed death, in order to multiply the number of cells of that group uncontrollably. Mutations occur during the natural life of an individual, induced by chemical, environmental and biological factors, but the most of their effects have no critical consequences as long as the equilibria in cellular populations are not compromised.

The errors in genetic transcription can avoid the G0 state in the cell cycle even in adverse to cell division conditions, promoting proliferation of a mutated phenotype. They can deeply modify the signal chain of promoter and inhibitor growth factors avoiding the generation of a DNA damage signal. The mutations can induce critical modifications of the ratio of anti-apoptotic versus pro-apoptotic proteins and this changes the cell resistance to programmed death, causing an enhanced survival when DNA errors occur. The accumulation of genetic errors permits genetic instabilities: a high frequency of variant cells can introduce abnormal motility features, easing invasions, or distort the differentiation program, raising the number of stem cells and then the proliferative potential.

Finally, cancer cells don't die we they have to, they exhibit an abnormal genetic variety, they avoid the quiescent state and they don't complete the maturation. All this features define the two most relevant properties of cancer: the uncontrolled reproduction and the damaging colonization of other tissues.

In the first stage its development a tumor experiences an avascular growth, [41]. It is able to proliferate using only the nutrients present where it is, limiting its size ($\sim 1mm$). This doesn't mean that tumor cells proliferate slowly, [41]. Apoptosis or tumor cell death occur for those parts unable to reach nutrient (mainly oxygen and glucose) and a high rate of death balances the generation of new cells, determining a steady state. Tumor growth depends on the survival under different deprivation conditions of oxygen supply, hypoxia, [41]. At this stage, the activities are mainly at intra- and inter-cellular level, defining the microscopic and mesoscopic scales.

A crucial step in tumor development toward malignancy is the angiogenesis switch, [41]. Tumors unable to induce angiogenesis remain dormant at microscopic or mesoscopic size, not developing a metastatic or lethal phenotype. Further mutations are needed allowing the neoplastic cells to become angiogenetic. The growth of new capillary blood vessels allows a constant uptake of nutrients and a great connection with the entire organism. At this stage tumor keeps on growing well connected and supplied and its size increases. Once a critical size is reached, the most insidious phases of cancer development may start, [60].

Invasion and metastasis processes involve the tissue level and their macroscopic effects are life-threatening. Local invasion kills the host by tissue compression and destruction, compromising the function of the tissues involved. Metastases induce distant organ injury, caused by cancer dissemination, and they change dramatically the therapeutic approach. Invasion is a highly efficient process relative to the ability of initiating new colonies: only 0.01% of circulating tumor cells initiate metastatic colonies, [60]. This is due to the extremely competitive and selective nature of the metastasis process, involving several tumor-host interactions and leading to the survival of a minor sub-population of the primary tumor. These are the terminal stages of the pathology, in which the high complexity of the disease is fully developed and its multiscale nature is evident.

5.1.2 Models

From the mutations inside a cell to the metastasis process involving tissues and organ far from the primary tumor, cancer development explore a wide range of

scales. The mechanisms involved in such a complex disease are not only numerous but often not yet understood. Moreover, a great variety in behaviours depending on the attacked organ, patients and environmental factors, increases the complexity of a model able to describe the cancer at once. For these reasons several approaches have been developed, focusing on the phases, aspects of the growth (a review in [24]) or on attacked organs (for lung [88], breast [42], brain [87]).

ODEs (*Ordinary Differential Equations*) models are featured by simplicity and low computational cost and, according to this they are currently the most used ones in clinical applications. They describe integral properties of the tissue evolution, generally the total size of the population and the nutrients dynamics. In early sixties, in her pioneering work, Laird ([56]) was able to fit tumor dynamics in simple experiments using a model describing the growth of a population in a confined space, where the availability of nutrients is limited, the Gompertz's model, [46]:

$$\dot{n}(t) = \alpha \log(\frac{K}{n(t)})n(t) \tag{5.1}$$

where $n(t)$ is the number of individuals at time t and K is the constant maximum size that can be reached with the available nutrients, the *carrying capacity*. More basic models were considered too, in order to model the growth of small tumors, for instance, the exponential and logistic ones. In the the work of d'Onofrio [35] a review of a family of models taking into account more complex dynamics can be found.

ODEs models can be easily used to couple the population dynamics and the therapeutic treatment. In Hahnfeldt [50], the Gompertz's model for tumor volume is considered with a variable carrying capacity K, defined as the effective vascular support provided to tumor. A new ODE for K is then introduced, being V the tumor volume:

$$K'(t) = -\lambda K + bV - dKV^{2/3} - eKg(t) \tag{5.2}$$

where the first term on the right hand side accounts for spontaneous loss of functional vasculature, the second and the third ones for stimulatory and inhibitory capacity of the tumor upon the inducible vasculature and the last one is the inhibition of the tumor vasculature due to administrate inhibitors (chemotherapy). However, ODEs models have an important drawback: integral quantities cannot describe space effects of tumor growth.

Spatial evolution can be considered using PDEs (*Partial Differential Equations*) models: discrete cell-based models and continuous models have been proposed, being able to cover different ranges of the intrinsic multiscale nature of cancer.

Discrete models are suitable for reproducing cancer evolution at the microscopic and mesoscopic scales: early tumor genesis and mutation effects are phenomena whose properties vary on scale comparable to the cell size. These models divide into lattice-based models (cellular automata models) and lattice-free models (agent models). In the former the cells are arranged on a lattice

and each node is associated to a number of states, [78]:

$$\mathbf{w} = \{\mathbf{x}, \mathbf{v}, \mathbf{u}\} \tag{5.3}$$

being $\mathbf{x}$ the position of the cell, $\mathbf{v}$ its velocity and $\mathbf{u}$ its internal biological state (position in cell cycle, interaction with local biochemical environment,...). Because of the correspondence between model cells and biological cells, considering heterogeneous population it is easy and it is an advantage of automata models, [78]. On the other hand, modelling cell motion and proliferation is harder. Usually a set of probabilities for proliferation, quiescence and movement is provided for each cell and rules are determined assuming interactions between a cell and its neighbours. Once the decision to proliferate or to move has been taken, the new cell or the moving one has to be put in extra space and this is one of the main problems for these models, implying numerical artefacts leading to unrealistic cell configurations. Deformable lattice, [78] and Lattice Gas Cellular Automaton model, [34] get over this matter.

The agent-based models are mathematically more complicated. They are lattice-free and, therefore, the cells (usually spheroids) can move in the space according mechanical laws and without grid constraints. Central potentials are often used to model cell interactions (mitosis appears in a more realistic description), [36], and pure random movement is assumed modelling dynamics through a Monte Carlo model. This allows these models to accurately fit *in vitro* experiments. In [70] a generalization of the classical methods of the kinetic theory of gases is used to model competition between tumor and immune cells, specifically Boltzmann formulation allows mean field description. Another approach is proposed in [75]: a cell-based model able to explicitly reproduce the cell membrane by an immersed boundary method allows to define more realistic cell interactions at the cell membrane receptors level.

However, the agent models are, like the other discrete models, computationally expensive and treating complex *in vivo* system attaining 10^{11} cells becomes unfeasible, even if efforts to reduce their computational cost have been made, inspiring the development of the hybrid models.

Continuous PDEs models reproduce the evolution of cell populations at the macroscopic tissue scale, using cellular densities. An exhaustive review can be found in [63]. These models describe the dynamics of a certain number of different cellular populations through continuity equations. Every cellular phenotype is depicted by a volumic density, ϕ_i and its specific mass is considered constant in time. The mass conservation equation for the i-th cellular species, therefore, reads:

$$\partial_t \phi_i + \nabla \cdot J_i = B_i - D_i + T_i \tag{5.4}$$

where J_i is a flux function and B_i, D_i and T_i are the birth, death and transition rates for the i-th phenotype. In a reference volume the number of cells of a certain population varies because of a flux across the boundary of the volume and some source terms.

Continuous models may differ in the choice of J_i defining the nature of the cell movement and, in case, the relationship between the populations and between a population and the environment.

A reaction diffusion model, like in [87], can be obtained choosing

$$J_i = D^{(i)} \nabla \phi_i + \sum_k^{N_p} V_{ik} \phi k \tag{5.5}$$

where $D^{(i)}$ is the diffusivity tensor and V_{ik} is the interaction matrix.

Otherwise,

$$J_i = \vec{v} \phi_i \tag{5.6}$$

where $\vec{v}$ is the velocity of the i-th phenotype, renders a transport. In [77] and in [4] the constrained mixture assumption is made, $\vec{v}_i = \vec{v} \quad \forall i$, also called the passive motion assumption, so that all the phenotypes don't undergo any relative motion or friction. The velocity introduced has to be characterize by constraints in models involving transport phenomena.

The saturation hypothesis is commonly assumed ([10]):

$$\sum_k^{N_p} \phi_k = 1. \tag{5.7}$$

As transitions, T_i sum to zero, this assumption implies a volume source role for mitosis, B_i and a volume sink role for death, D_i:

$$\nabla \cdot \vec{v} = \sum_i B_i - D_i + T_i. \tag{5.8}$$

This means that after cell division two daughter cells occupy a greater volume relative to the mother cell, pushing the neighbouring cells.

Another constraint on $\vec{v}$ is needed for the mechanical closure of the system. The momentum conservation equation

$$\partial_t (\phi_i \vec{v}) + \nabla \cdot (\phi_i \vec{v} \otimes \vec{v}) = \nabla \cdot \sigma_i + \phi_i \vec{b_i} \tag{5.9}$$

where σ_i is the stress tensor for the i-th phenotype and $\vec{b_i}$ is the body force acting on it, can be completed by several constitutive equations, generating as much models, ([79], [25], [5]). In [6], for instance, the tumor is supposed to be in quasi-steady conditions, resulting in $\nabla \cdot \sigma_i = 0$.

In ([4], [63]) a generalized potential flow model is considered, letting the tissue to behave as a flow in a porous medium and reducing the momentum equation to a Darcy law:

$$\vec{v} = -k \nabla \Pi \tag{5.10}$$

where k_i is the permeability tensor and Π is the scalar potential function for the velocity.

In order to couple the populations equations to the environment a description of nutrients evolution has to be provided. Generally speaking, the nutrients are all the substances regulating the tumor growth, but considering the complex assortment is unfeasible. The oxygen, the glucose and the H^+ ions are classically taken into account for the role in cell energetics of the first two and for the acidosis effects the third can cause, [78].

The nutrients dynamics is usually governed by a reaction diffusion equations for the concentration of the k-th species, c_k:

$$\partial_t c_k - \nabla \cdot (D_k(\phi_i)\nabla c_k) = C(\phi_i, c_k) + S(\phi_i, c_k) \tag{5.11}$$

where D_k is the diffusivity tensor, function of the phenotype populating the domain, C is the k-th species consumption rate for the i-th phenotype.

Another important equation is needed to specify the relationship between the populations and the availability of nutrients. Such an equation is usually introduced to control the mitosis process by a switch-like mechanism:

$$B_i(\phi_j) = \nu_i H(c_k - \bar{c}_k) \tag{5.12}$$

where $H(\cdot)$ is the Heavieside function, B_i is the mitosis rate, ν_i is the inverse of the mitosis time scale and $\bar{c}_k$ is a threshold. It states that under a certain threshold of the k-th nutrient the mitosis is switched off, driving the i-th phenotype to quiescence or apoptosis, otherwise the phenotype can proliferate. Surely, boundary conditions have to be imposed both for the cell population equations and for the nutrient equations according to the nature of the organ.

More sophisticated continuous models arise form the consideration of the cell cycle (age-structured models, [76], [77]), other constitutive laws (for instance, using a thermodynamic argument, [31]), not diffused phases (interface models, [64]) considering the tumor interface and imposing curvature-based conditions on it, complex processes as angiogenesis ([18], [7] for a review) or systemic treatments like chemotherapy ([76]), introducing further nutrients equations.

The *hybrid models* represent the effort to take into account the intrinsic multiscale nature of tumor growth. Several approaches have been introduced in literature, trying to take advantage of the strong points of continuous and discrete models. In [74] the authors extended discrete automata models to the continuous limit to study how microscopic movement rules translate into parameters of macroscopic continuous partial differential equations. Other approaches use the two classes of descriptions for the different constituents of the model: in [43] continuous fields are used to represent nutrients and discrete fields for the cell populations, coupling the description but avoiding the mix; conversely, in [55] a discrete agent based model is used to describe the dynamics of the most relevant phase, the proliferating cells, and continuous fields are introduced for the other constituents, the quiescent phase, for instance; therefore, phase transitions involve least square projections when the continuous phase begin to proliferate in order to create new discrete proliferating individuals. Different criteria may be applied in transitions from one description to another one: in [16] both descriptions are used for all the components of the model, choosing one or the other by means of hypoxia and density arguments.

Some of these hybrid models don't improve the computational cost which remains as high as the cost of the discrete models embedded in, [43]. Some others allow a significant reduction, if they are applied to specific stage of tumor growth [55]. However, they successfully enlarge the range of the scales the discrete and continuous models can handle separately.

5.2 The two-species Darcy model

The present application has to be intended in continuation with the works in [29] and [62]. The methods developed in these works are aimed at being applied in practical, clinical situations. As we have already pointed out, classically, ODEs models are used for clinical applications, but we have also remarked that using integral quantities, they are not able to deduce any information about the shape and the location of the tumor.

Therefore, in order to propose methods accounting for the spatial aspects of tumor growth, the authors in [29] have chosen to operate in the framework of PDEs models and in [62] the choice of a continuous model is motivated. Medical imagery is the main (in quantity) source of information about the dynamics of the patient pathology and it is able to explore human body at tissue level. This means that the primary information about tumor is collected at the macroscopic scale of the evolution, motivating the choice of a continuous PDEs model. Moreover, the clinical detection of the tumor by means of medical imagery happens when its size is already some millimetres, so that the propagation is at the macroscopic scale, reinforcing the use of continuous models.

Among the continuous models in the previous section, [63] and [20] give an accurate insight in biological phenomena involved in tumor growth, but both [29] and [62] explain why introducing such models in clinical applications is difficult.

The continuous models are not able to represent the microscopic and mesoscopic scales of cancer development. The effects at these scales have to be lumped in the parameters of a macroscopic model. Because of the complex nature of cancer, modelling has a phenomenological nature. Mainly for *in vivo* systems, the models in literature have been proposed and validated fitting experiments rather than inferring macroscopic equations from first principles. As a result, the parameters often have not clear biological and physical meaning and they can rarely be directly measured.

The aim of methods proposed in [29] and [62] is to efficiently identify these parameters, exploiting data from medical imagery, by means of inverse problems with the final objective of obtaining a prognostic model. Therefore, introducing complex continuous models with a huge number of free parameters would render the task unaffordable. The model adopted is a trade-off between the simplicity and the accuracy, [29], giving a reasonable description of the phenomenon and defining a computationally achievable identification problem.

The model at issue is the two species Darcy-type model. It describes the dynamics of three phases, considering the tissue as a flow in a porous isotropic non-uniform medium. Far from being as sophisticated as the models introduced in the previous section ([76], [31], [64], [18]), it disregards biological mechanisms like cell cycle, angiogenesis, focusing on the main features of tumor growth. Proliferation, oxygen consumption, undergoing hypoxia, quiescence are the mechanisms the model can easily take into account, even angiogenesis effects could be modelled choosing the oxygen parameters and verifying *a posteriori* the efficiency of the model, [62].

The three populations considered in the model are described by cellular volumic densities. P is the number of proliferating cells per unit volume: these are the diving cells responsible for the tumor growth; Q is the number of qui-

escent cells per unit volume: their quiescence state is regulated by the presence of oxygen; S is the number of healthy cells per unit volume: their metabolism is not fast as the proliferative cell one, [76], and they undergo compression due to the action of the other phases. Their dynamics is modelled by mass conservation equations as in (5.4), choosing the fluxes J_i in order to render transport, namely:

$$\frac{\partial P}{\partial t} + \nabla \cdot (\vec{v}P) = (2\gamma - 1)P + \gamma Q \tag{5.13}$$

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\vec{v}Q) = (1 - \gamma)P - \gamma Q \tag{5.14}$$

$$\frac{\partial S}{\partial t} + \nabla \cdot (\vec{v}S) = 0 \tag{5.15}$$

The passive motion assumption has been assumed, [4], then $\vec{v}_P = \vec{v}_Q = \vec{v}_S = \vec{v}$. The scalar function γ is called hypoxia threshold and it depends on oxygen concentration. Its explicit definition will be introduced later on, when the oxygen equation will be detailed. At the moment the equation (5.15) for healthy cells only shows the slower healthy cell metabolism relative to the proliferating one, since the effect of compression needs further hypotheses.

Equations (5.13) and (5.14) have right hand sides we need to discuss. We write again the equation for P to isolate the contributions to the dynamics:

$$\frac{\partial P}{\partial t} + \nabla \cdot (\vec{v}P) = \underbrace{\gamma P}_{\text{cell division}} + \underbrace{(\gamma - 1)P + \gamma Q}_{\text{transitions}} \tag{5.16}$$

The first term in the right hand side describes cell division and it is a source for the proliferating phase. The other two terms account for transitions between the proliferative and the quiescent phases and they can also be found in equation (5.14) with opposite sign.

The hypoxia threshold γ varies in $[0, 1]$ according to the oxygen concentration. As long as enough oxygen is available and $\gamma = 1$ in equation (5.16) we have production (γP) of proliferative cells by cell division and transition from quiescent to proliferative phase ($+\gamma Q$ in (5.16) and $-\gamma Q$ in (5.14)). Consuming oxygen, the tumor experiences several degrees of hypoxia, as the matter of fact if $\gamma < 1$ some proliferative cells keep on dividing, but some others enter the quiescence state (terms $(\gamma - 1)P$ in (5.16) and $(1 - \gamma)P$ in (5.14)). For very low level of oxygen γ can be zero: in this case no cell division is possible and no quiescent cell can start to proliferate and all the tumor cells become quiescent. The γ function is one of those parameters the works [29] and [62] want to estimate. Its pure phenomenological nature prevents any measure in experiments. Several complex mechanisms, like angiogenesis or gene activation behaviour in reply to hypoxia signals [76], are lumped in it and hence it has to be identified.

We introduce the hypothesis of saturated flow, commonly used in mixture theory, [10], which means $P + Q + S = 1$. This assumption has an important mechanical consequence. If equations (5.13),(5.14),(5.15) are summed up, the result of the sum is

$$\nabla \cdot \vec{v} = \gamma P \tag{5.17}$$

This equation says that under saturated flow assumption the mitosis process is a volume source for the flow and according to the G2 phase in the cell cycle (see section 5.1.1) this reasonable being the division anticipated by the cell growth.

Mechanical closure of the system needs a further equation for $\vec{v}$ and here it is provided by the hypothesis of quasi-steady flow in porous medium with variable porosity, k, giving a Darcy-type law which reads:

$$\vec{v} = -k(P,Q)\nabla\Pi \tag{5.18}$$

The scalar function Π is the potential for the velocity, playing the role of a pressure (see [4]). The simplest phenomenological law for the porosity field k is a linear relation with volumic densities of the tumor phases:

$$k = k_1 - (k_1 - k_2)(P + Q) \tag{5.19}$$

where k_2 is the constant porosity in the tumor tissue and k_1 the constant porosity in the healthy one.

Among the nutrients we choose the oxygen, giving it the major role in regulating the cells activities. Its evolution is regulated by a reaction-diffusion equation for its concentration, C:

$$\frac{\partial C}{\partial t} - \nabla \cdot (D(P,Q)\nabla C) = \lambda(C_{max} - C)S - \alpha PC - 0.01\alpha QC \tag{5.20}$$

where $D(P,Q)$ is the diffusivity, λ is the oxygen production rate for healthy cells and α is the oxygen consumption rate for proliferating cells. The first term in the right hand side accounts for oxygen healthy cells production and it prevents oxygen to reach concentration higher than C_{max}. The second and the third terms describe the oxygen consumption for tumor cells. Again, a linear phenomenological law for the parameter D is chosen in such a way the diffusion of oxygen is different in healthy and tumor tissues:

$$D = D_{max} - K(P + Q) \tag{5.21}$$

The hypoxia threshold function γ is modelled by a mollified Heavieside function of the oxygen concentration:

$$\gamma = \frac{1 + \tanh(R(C - C_{hyp}))}{2} \tag{5.22}$$

where R is a coefficient and C_{hyp} is the hypoxia oxygen concentration threshold. It works as a tempered switch for the mitosis process, also regulating transition between the two tumor phases. The rapidity of the step function can be chosen, changing the value of the coefficient R. Its mollified nature allows the coexistence of proliferation and transition to the quiescence state for the P phase and of two-way transitions for the Q phase.

Dirichlet or Neumann boundary conditions have to be included according to the physics of the system and the clinical case for both oxygen and pressure. For instance, the presence of a vessel can be rendered imposing constant oxygen concentration in a sub-domain.

In the present application we are interested in imposing that no mass can leave the domain, therefore in imposing homogeneous Neumann boundary conditions for both oxygen and pressure. But this lays down a modification of the equation for the divergence of the velocity, in order to have a well posed problem:

$$\nabla \cdot \vec{v} = \gamma(C)P - \frac{\int_\Omega \gamma P \, d\Omega}{\int_\Omega (1 - P - Q) \, d\Omega}(1 - P - Q) \tag{5.23}$$

As we have anticipated, this implies, from a mechanical point of view, that tumor growth compress the healthy tissue and equation (5.15) can no longer holds, being the healthy cells no longer governed by a homogeneous transport.

5.3 The numerical framework

In this section we provide some details about the numerical techniques used to perform three-dimensional simulations of the model introduced in section 5.2. The numerical methods adopted are implemented in *eLYSe* ([37], available at [3]), a computing platform developed for tumor growth, microfluidics and complex fluids simulations.

The problem provided by the two species Darcy-type model consists in a system of coupled elliptic, hyperbolic and parabolic PDEs, a Darcy-type law and a saturation condition (six equations) for the six fields $P, Q, S, \Pi, C, \vec{v}$ with six parameters $k, D, \gamma, \alpha, \lambda, C_{max}$ (formally nine constant parameters $k_1, k_2, D_{max}, K, R, C_{hyp}, \alpha, \lambda, C_{max}$) on Ω and with homogeneous Neumann boundary conditions for pressure and oxygen concentration on $\partial\Omega$. As a whole it reads:

$$\frac{\partial C}{\partial t} - \nabla \cdot (D(P,Q)\nabla C) = \lambda(C_{max} - C)S - \alpha PC - 0.01\alpha QC \qquad \text{in } \Omega \tag{5.24}$$

$$\frac{\partial C}{\partial \vec{n}} = 0 \qquad \text{on } \partial\Omega \tag{5.25}$$

$$\frac{\partial P}{\partial t} + \nabla \cdot (\vec{v}P) = (2\gamma(C) - 1)P + \gamma(C)Q \qquad \text{in } \Omega \tag{5.26}$$

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\vec{v}Q) = (1 - \gamma(C))P - \gamma(C)Q \qquad \text{in } \Omega \tag{5.27}$$

$$P + Q + S = 1 \qquad \text{in } \Omega \tag{5.28}$$

$$-\nabla \cdot (k(P,Q)\nabla\Pi) = \gamma(C)P - \frac{\int_\Omega \gamma(C)P \, d\Omega}{\int_\Omega (1 - P - Q) \, d\Omega}(1 - P - Q) \qquad \text{in } \Omega \tag{5.29}$$

$$\frac{\partial \Pi}{\partial \vec{n}} = 0 \qquad \text{on } \partial\Omega \tag{5.30}$$

$$\vec{v} = -k(P,Q)\nabla\Pi \qquad \text{in } \Omega \tag{5.31}$$

$$\gamma = \frac{1 + \tanh(R(C - C_{hyp}))}{2} \tag{5.32}$$

$$D = D_{max} - K(P + Q) \tag{5.33}$$

$$k = k_1 - (k_1 - k_2)(P + Q) \tag{5.34}$$

where $\vec{n}$ is the outward normal unit vector of $\partial\Omega$. The space discretization is based on standard finite differences on Cartesian grid and the code implementation is parallel (PETSc and MPI, [14], [2]).

5.3.1 Domains

Before discussing the specific numerical methods used for every equation we want to briefly point out what kind of domains we deal with and to introduce the tools used to get a mathematical description of them.

Two domains are considered in the present application: a very simple and regular one, a sphere, and the quite irregular inside of a lung. For both of them we need a description which allows us to evaluate certain geometrical features, like the position relative to the grid and the normals of the boundary surface, in an efficient way from the parallel implementation point of view.

As we have already remarked in sections 4.3.1 and 4.5.2 the level set function is particularly suitable for parallel implementation, but, if for a sphere it can be easily derived, for a lung we need to do something for good measure.

For a start we need to collect information about the shape of a lung and the medical imagery is a good source of information in this sense. Generally speaking, it is a fundamental tool in medical procedures and in the present case it allows us to capture the shape of a realistic lung.

Right form the beginning, it is better to say that we are not interested in the complex internal structure of a realistic lung, for the present application we focus our attention on the boundary of this organ, in order to limit the geometric complexity of the problem. But, the techniques we rapidly introduce in the following are able to catch much more details than we want to consider.

Several medical imaging techniques are available nowadays, their use is usually chosen according to the anatomical part of the body that technique fit better.

Magnetic Resonance Imaging (MRI) is suitable for brain, spinal cord and muscles. Its resolution is approximately 3mm.

Positron Emission Tomography (PET) is heavily used in clinical oncology for its ability in detecting metastases. Its precision is lower compared to MRI ($\sim$ 5mm).

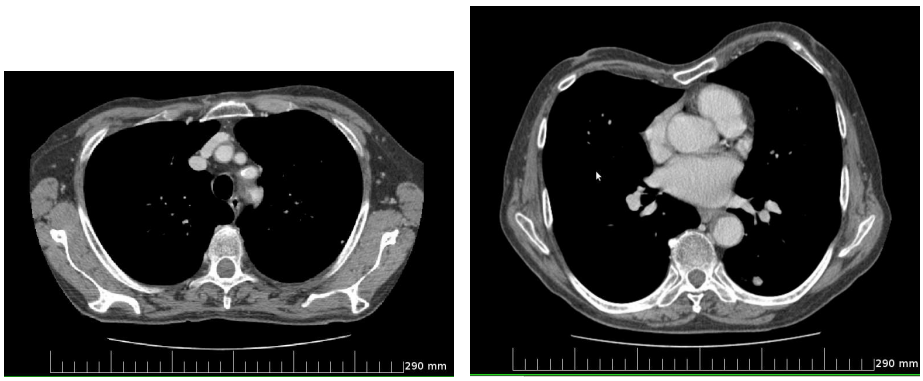
Single-photon emission computed tomography (SPECT) is good for liver and in neuro-oncology. It offers the same resolution of PET.

Computed Tomography (CT-scans) is used for the diagnosis of almost all the lung diseases and for several other organs, like liver, kidney, pancreas and colon. Its resolution is high relative to the other techniques attaining a precision of 1mm.

All these techniques allow to obtain three-dimensional images, exploiting different physical principles (see [22]). For the present application we use CT-scans provided by medical doctors.

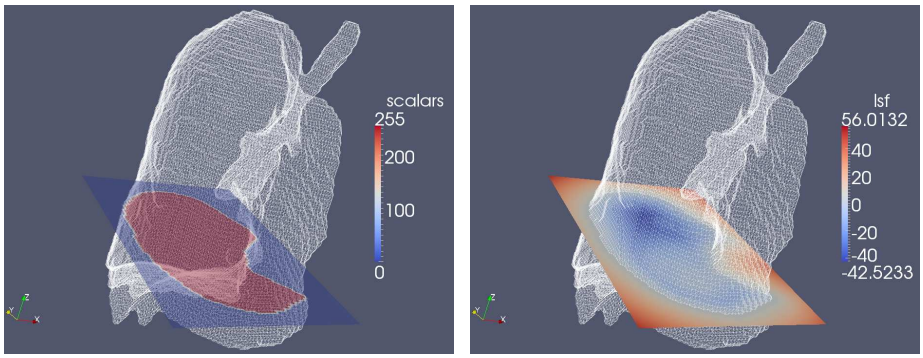
However, the images from CT are not ready to be used in simulations, since they are a collection of two-dimensional grey scale parallel scans of the patient chest taken along the longitudinal axis. A couple of this scans is given in Figure 5.2.

In order to extract a three-dimensional description of the lung boundary a segmentation procedure is applied. Specifically, the software used for this purpose has been developed by M. Specklin and R. Bahègne at the Insitut de Mathèmatiques de Bordeaux, using the Insight Segmentation and Registration Toolkit (ITK, see [91] and [1]). The segmentation method implemented is a standard threshold method. Moreover, CT-scans of lung have good contrast and resolution, therefore segmentation algorithms provide quite good results.



(a) CT-scan of the upper part of the chest (b) CT-scan of the central part of the chest

Figure 5.2: CT-scans. Courtesy Institut Bergonié, Bordeaux



(a) Binary mask (b) Level set function

Figure 5.3: Segmented Lung

However, the three-dimensional fields obtained by segmentation are binary fields (masks) on Cartesian grid: usually for the points inside the lung boundary the value is 255 while for the points outside the value is 0, see Figure 5.3a. The boundary of the lung is then defined as the locus of midpoints between two grid nodes with different values of the mask.

Therefore, level set function is computed, initializing the first neighbours to the exact signed distance and applying the Fast Marching Method (FMM, see [82] and [84]) twice to compute signed distance far from the interface. The code implementing this method has been developed by Olivier Saut and it is available in eLYSe, [37]. Finally, the level set function for the lung is obtained (Figure 5.3b). Examples in Figure 5.3 show a lung in a $54 \times 86 \times 100$ domain.

We remark that the domain Ω (a lung or a sphere) is now embedded in a larger one by the use of a level set function description and that its boundary $\partial\Omega$ appears as a close interface immersed in a regular Cartesian grid: exactly what we need to apply the method introduced in the previous chapter.

5.3.2 The elliptic irregular domain problem

We go back on the equations focusing on the elliptic problem for pressure, defined by equations (5.29) and (5.30), and on the parabolic problem for the oxygen concentration, defined by equations (5.24) and (5.25). For these equations we want to exploit the method for elliptic interface problem introduced in Chapter 4. The main idea is to embed the Ω domain in regular Cartesian grid defining a new larger domain Ω' and to impose the Neumann boundary condition on $\partial\Omega$ in a penalty method spirit ([11], [9]).

For the sake of simplicity and visualization, we propose a two-dimensional example in order to explain how the interface method can be used in solving an elliptic irregular domain problem.

Let us consider the following two-dimensional elliptic problem:

$$\nabla \cdot (k\nabla u) = f \quad \text{in } \Omega \tag{5.35}$$

$$\frac{\partial u}{\partial \vec{n}} = 0 \quad \text{on } \partial\Omega \tag{5.36}$$

$$\tag{5.37}$$

with

$$f = \operatorname{sgn}(x) A e^{-\frac{(|x|-x_c)^2+y^2}{b}} - \operatorname{sgn}(x) A e^{\frac{x_c^2+y^2}{b}} \quad \text{and} \quad k = 1 \quad \text{in } \Omega, \tag{5.38}$$

$$\Omega : \phi(x,y) < 0,\ \phi = \sqrt{x^2+y^2} - 1 \tag{5.39}$$

A solution to this problem exists only if the compatibility condition, relating the source of the Poisson equation and the Neumann boundary condition, is satisfied. It is easy to verify that the integral of f over Ω is equal to zero and the compatibility is fulfilled. The solution u is defined short of a constant.

We embed the domain Ω in a rectangular domain $\Omega' = [-3,3] \times [-3,3]$, extending the definition of f, u and k to f', u' and k', imposing homogeneous transmission conditions on $\partial\Omega$ and stating a new interface problem:

$$\nabla \cdot (k'\nabla u') = f' \quad \text{in } \Omega' \tag{5.40}$$

$$[\![u']\!] = 0 \quad \text{on } \partial\Omega \tag{5.41}$$

$$[\![k'\frac{\partial u'}{\partial \vec{n}}]\!] = 0 \quad \text{on } \partial\Omega \tag{5.42}$$

$$\frac{\partial u'}{\partial \vec{n}'} = 0 \quad \text{on } \partial\Omega' \tag{5.43}$$

with

$$f' = \begin{cases} f & \text{in } \Omega \\ 0 & \text{in } \Omega' \setminus \Omega \end{cases}, k' = \begin{cases} k & \text{in } \Omega \\ \tilde{k} & \text{in } \Omega' \setminus \Omega \end{cases}, u' = \begin{cases} u & \text{in } \Omega \\ \tilde{u} & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.44}$$

and where $\vec{n}'$ is the normal vector of $\partial\Omega'$. The compatibility conditions is still satisfied, extending f to zero in $\Omega' \setminus \Omega$. We remark that equation (5.42) can be re-written as follows:

$$\frac{\partial u}{\partial \vec{n}} = \frac{\tilde{k}}{k}\frac{\partial \tilde{u}}{\partial \vec{n}} \tag{5.45}$$

Therefore, choosing $\tilde{k}$ small enough, we can force $\frac{\partial u}{\partial \vec{n}}$ to be sufficiently small to be considered zero and we can solve and approximate the irregular domain problem by an interface problem. According to this, the method introduced in the previous chapter can be used to find an approximation of u in Ω. However, such approximation depends on the value of the normal derivative of $\tilde{u}$ on $\partial\Omega$.

In Table 5.1 we propose a small study of the normal derivatives of u and $\tilde{u}$ on $\partial\Omega$ varying the parameters A and $\tilde{k}$ of the two-dimensional interface problem we have just proposed. We choose $x_c = 0.5$ and $b = 0.01$ for this test. Far from being a proof of goodness for the procedure, it shows that carefully choosing the value of $\tilde{k}$ we can impose a good approximation of homogeneous Neumann conditions on the boundary of an irregular domain.

			$\tilde{k}$		
			1	10^{-6}	10^{-9}
A	1	$\max(\lvert\frac{\partial u}{\partial \vec{n}}\rvert)$	10^{-3}	10^{-8}	10^{-11}
		$\max(\lvert\frac{\partial \tilde{u}}{\partial \vec{n}}\rvert)$	10^{-3}	10^{-2}	10^{-2}
	100	$\max(\lvert\frac{\partial u}{\partial \vec{n}}\rvert)$	10^{-1}	10^{-6}	10^{-9}
		$\max(\lvert\frac{\partial \tilde{u}}{\partial \vec{n}}\rvert)$	10^{-1}	10^{0}	10^{0}
	10000	$\max(\lvert\frac{\partial u}{\partial \vec{n}}\rvert)$	10^{1}	10^{-4}	10^{-7}
		$\max(\lvert\frac{\partial \tilde{u}}{\partial \vec{n}}\rvert)$	10^{1}	10^{2}	10^{2}

Table 5.1: Normal derivatives of the solution on $\partial\Omega$, varying A and $\tilde{k}$

A solution for the case $A = 100$ and $\tilde{k} = 10^{-6}$ is proposed in Figure 5.4. The solution contour lines in Ω show their approximate perpendicularity relative to $\partial\Omega$.

We impose continuity of the solution on $\partial\Omega$ (see equation (5.41)) and this is surely a limit in the case of strong sources. On the other side, considering the possibility of using the solution jump to control the normal derivative of the solution outside the original domain is a chance for future improvements. As we will see in section 5.4 sources for pressure and oxygen concentration equations are not so strong and they give us the chance to use this procedure with reasonable small $\tilde{k}$.

The three-dimensional Poisson problem (equations (5.29), (5.30) in the model) for the pressure Π is embedded in a new larger domain we can call Ω', extending

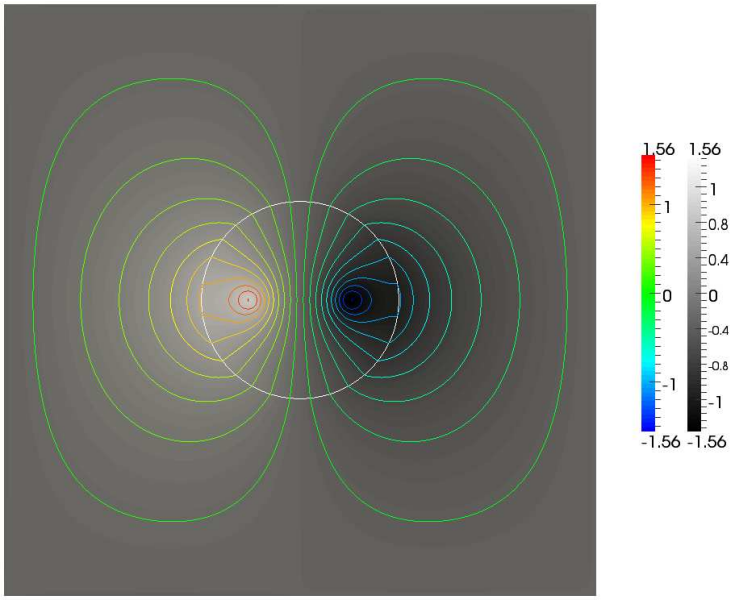


Figure 5.4: Two-dimensional elliptic irregular domain problem test. $A = 100$, $\tilde{k} = 10^{-6}$, $x_c = 0.5$, $b = 0.01$

$k(P,Q)$, Π and the right hand side of equation (5.29) to $\Omega' \setminus \Omega$:

$$-\nabla \cdot (k'(P,Q)\nabla \Pi') = f' \qquad \text{in } \Omega' \tag{5.46}$$

$$[\![\Pi']\!] = 0 \quad \text{on } \partial\Omega \tag{5.47}$$

$$[\![k'(P,Q)\frac{\partial u'}{\partial \vec{n}}]\!] = 0 \quad \text{on } \partial\Omega \tag{5.48}$$

$$\frac{\partial \Pi'}{\partial \vec{n}'} = 0 \qquad \text{on } \partial\Omega' \tag{5.49}$$

with

$$f' = \begin{cases} \gamma(C)P - \frac{\int_\Omega \gamma(C)P \, d\Omega}{\int_\Omega (1-P-Q) \, d\Omega}(1-P-Q) & \text{in } \Omega \\ 0 & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.50}$$

$$k'(P,Q) = \begin{cases} k(P,Q) & \text{in } \Omega \\ \tilde{k} & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.51}$$

$$\Pi' = \begin{cases} \Pi & \text{in } \Omega \\ \tilde{\Pi} & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.52}$$

C, P, Q have to be meant at the previous time step. The discretization is performed using formulas in section 4.4.

As far as the oxygen concentration equation is concerned, a first order time discretization is performed using an explicit scheme for the reaction part of the equation and a implicit scheme for the diffusion term. The semi-discretized (in

time) problem reads:

$$-\nabla \cdot (D(P^n, Q^n)\nabla C^{n+1}) + \frac{C^{n+1}}{\Delta t} = f \qquad \text{in } \Omega \tag{5.53}$$

$$\tag{5.54}$$

with

$$f = \frac{C^n}{\Delta t} + \lambda(C_{max} - C^n)S^n - \alpha P^n C^n - 0.01\alpha Q^n C^n \tag{5.55}$$

and homogeneous Neumann conditions on Ω. At each time step the parabolic problem for the oxygen concentration reduces to an elliptic irregular domain problem. According to this, we decide to solve equation (5.53) adopting the same approach used for pressure equation. Hence, the new problem for oxygen concentration reads:

$$-\nabla \cdot (D'(P^n, Q^n)\nabla C'^{n+1}) + \frac{C'^{n+1}}{\Delta t} = f' \qquad \text{in } \Omega' \tag{5.56}$$

$$[\![C'^{n+1}]\!] = 0 \quad \text{on } \partial\Omega \tag{5.57}$$

$$[\![D'(P^n, Q^n)\frac{\partial C'^{n+1}}{\partial \vec{n}}]\!] = 0 \quad \text{on } \partial\Omega \tag{5.58}$$

$$\frac{\partial C'^{n+1}}{\partial \vec{n}'} = 0 \qquad \text{on } \partial\Omega' \tag{5.59}$$

with

$$f' = \begin{cases} \frac{C^n}{\Delta t} + \lambda(C_{max} - C^n)S^n - \alpha P^n C^n - 0.01\alpha Q^n C^n & \text{in } \Omega \\ 0 & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.60}$$

$$D'(P^n, Q^n) = \begin{cases} D(P^n, Q^n) & \text{in } \Omega \\ \tilde{D} & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.61}$$

$$C'^{n+1} = \begin{cases} C^{n+1} & \text{in } \Omega \\ \tilde{C} & \text{in } \Omega' \setminus \Omega \end{cases} \tag{5.62}$$

A slightly modified version of the elliptic interface problems method in section 4.4 can be used. Equations (4.36) and (4.37) have to account for the new term $\frac{C'^{n+1}_{i,j,k}}{\Delta t}$ responsible for a constant contribution to diagonal matrix elements in elliptic operator rows.

Both pressure and oxygen concentration solutions in $\Omega' \setminus \Omega$ are discarded, having no physical meaning.

Equation (5.31), the Darcy-type law for velocity, is approximated using central second-order finite differences scheme in all the nodes in Ω. For those grid points too close to the boundary $\partial\Omega$ the values of the pressure in the intersection points are used. For the sake of coherence with the topic of the section we point out that the velocity field is not defined outside the original domain Ω, having no meaning and being not necessary from a numerical point of view.

5.3.3 Transport

Transport equations (5.26) and (5.27) for the two tumor cell populations are computed using classical ENO2 method, ([51], [32], [33]) and first order explicit scheme in time. The divergence part of the conservation equations is taken into account as if it were a source of the transport equation, computed by means of an exponential term.

For the phenotype P the semi-discretized equation reads:

$$P^{n+1} + \Delta t(\vec{v}^n \cdot \nabla P^n) = P^n + \Delta t(-P^n(\nabla \cdot \vec{v})^n + (2\gamma^n - 1)P^n + \gamma^n Q^n) \tag{5.63}$$

Once the phases P and Q are computed at time step $n+1$ with this scheme, the healthy cells density, S^{n+1}, is obtained using the saturated flow condition, 5.28. The resulting discretization scheme is not conservative and conditionally stable. Therefore, a CFL condition has to be imposed and time step is updated at each time iteration according to this prescription.

5.3.4 Discussion

The overall numerical method is first order in time and second order in space. Time step limitation come from the transport part of the model in order to keep the adopted scheme stable.

The methods for transport equation are a weak point for the overall method. The explicit first order scheme for time integration are poorly accurate, diffusive and, furthermore, not conservative. On the other side, their parallel implementation is straightforward and this allowed us to focus on the application of the method for elliptic interface problem in the framework of the tumor growth model.

On the road toward realistic applications this aspect has to be improved, considering the implementation of more accurate, high order schemes which can limit the mass loss effects and offer results less affected by poor numerical accuracy. High order in time Runge-Kutta schemes (see [47] for TVD schemes) and high order in space WENO5 schemes, [90], can do the job and their implementation is in the schedule of method improvement.

As far as the schemes for pressure and oxygen equation are concerned, we believe that the use of the interface method to impose homogeneous Neumann boundary conditions has to be deeply investigated and that the following application is just the first phase of this survey. A way to control the normal derivative of the solution on the outer side of the original domain is necessary, in order to apply the method to a wider range of situations and we believe that the solution jump and the source can be carefully chosen for this purpose. The time scheme for parabolic problem can surely be improved, introducing for example the Crank-Nicolson scheme,[30], to get a second order in time, once the transport will be second order in time too.

Despite all the improvements the method surely needs, the following application can show, at a quality level, if the interface method can play a role in elliptic irregular domain problem in the framework of three-dimensional tumor growth modelling, allowing a second order discretization in space up to the boundary of an impermeable domain.

5.4 Simulations

This section is devoted to the results of the simulations of the two species Darcy-type tumor growth model (equations (5.24)-(5.34)) performed in the numerical framework provided in section 5.3.

All the simulations have been conducted on the nodes of the High Performance Computing machine Fourmi at the "Plateforme Fédérative pour la Recherche en Informatique et Mathématiques en Acquitane" (PlaFRIM, see [3]). For both cases in the present work we used a cubic Cartesian grid of 64 processors.

The parameters are chosen in the range of the values used to construct the databases for identification problems ([29]). The choice promotes proliferation of tumor cells, inhibiting the quiescence state, even if the use of mollified unit step function for the hypoxia threshold allows two-way transitions among the phases in high oxygen concentration conditions too. In this sense, the consumption of oxygen is assumed small in order to ease the growth of the tumor nodule in a well oxygenated environment. In Figure 5.5, we provide the shape of the gamma function used in the simulations, with $C_{hyp} = 0.5$ and $R = 10$.

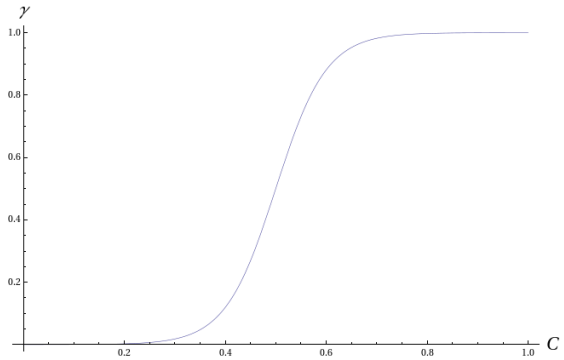


Figure 5.5: Hypoxia threshold, γ, for $C_{hyp} = 0.5$ and $R = 10$

We select two different geometries for the Ω domain and two different initial distributions for the tumor phenotypes P and Q. On the other hand the oxygen concentration at the beginning of every simulation is chosen uniform and equal to the maximum for both the cases, $C_\Omega = 1$ and $C_{\Omega' \setminus \Omega} = 0$.

In the following we call the sum of the two tumor species nodule. Initial nodules are localised in regular geometries saturated by only one species, i.e. P and Q assume the maximum density value in small, clearly non-overlapping, sub-domains. No special treatment of the nodule interface is considered, and no internal structure too. Nodules are simply defined as region where the sum of the tumor species is one at beginning. This has important consequence: since the initial size of the tumor mass is some grid steps, what we define analytically as a smooth sub-domain results in a strongly approximated cluster of cells, preserving the memory of this initialization during its evolution.

Spherical domain with spherical centred nodule. For this first simulation the domain Ω is a sphere in the origin of the axes with radius $\rho_\Omega = 0.4$

embedded in a domain $\Omega' = [-0.5, 0.5] \times [-0.5, 0.5] \times [-0.5, 0.5]$.

$$\bar{\Omega} \ : \ \phi \leq 0, \ \text{with} \ \phi = \sqrt{x^2 + y^2 + z^2} - \rho$$

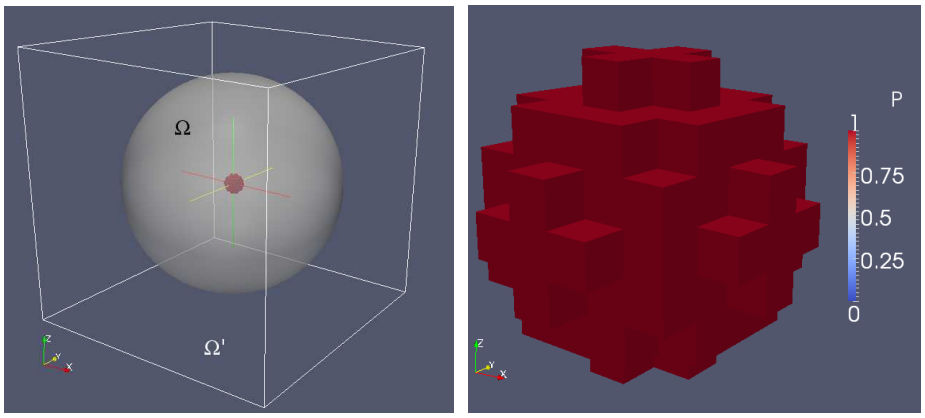
The model is discretized on a $100 \times 100 \times 100$ Cartesian grid.

The nodule is a small sphere with the same center and radius $\rho_{nod} = 0.04$, saturated with the proliferating phenotype P.

$$P = \begin{cases} 1, & \text{if } \sqrt{x^2 + y^2 + z^2} < 0.04 \\ 0, & otherwise \end{cases}$$

The quiescent phenotype, Q, is zero everywhere. Figure 5.6 shows the geometrical set-up and the initial nodule in red. Recalling the meaning of the parameters, we provide our selection:

- maximum oxygen concentration, $C_{max} = 1$,
- oxygen production rate, $\lambda = 0.1$,
- oxygen consumption rate, $\alpha = 1.98$,
- porosity of healthy and tumor tissue, respectively, $k_1 = 1$ and $k_2 = 1.2$,
- diffusivity of the healthy tissue, $D = 1$ and $K = 0.2$,
- diffusivity and porosity in $\Omega' \setminus \Omega$, $\tilde{k} = \tilde{D} = 10^{-6}$,
- CFL=0.45.



(a) Geometrical set-up (b) The nodule

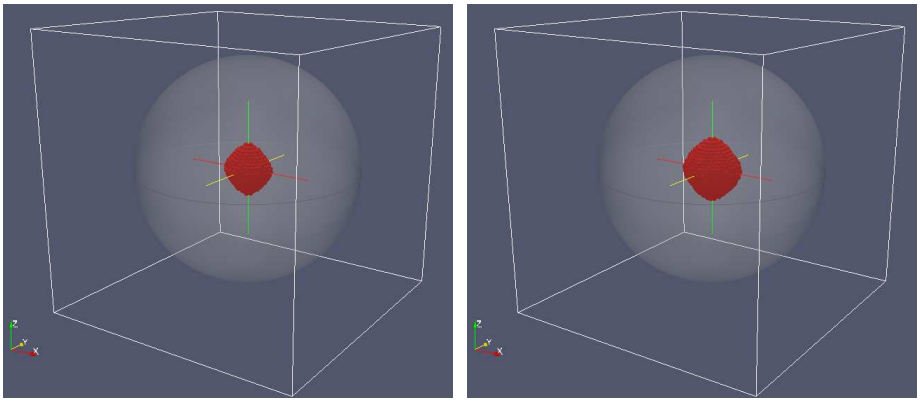
Figure 5.6: Geometrical set-up and initial nodule for the first simulation. The white box is the computational domain, Ω'; in transparent white Ω; in red the nodule.

In order to chose the values $\tilde{D}$ and $\tilde{k}$, some tests have been performed. Considering the two-dimensional example in the previous section, it seems a reasonable choice: the source terms in equations (5.50) and (5.60) have maximum values of order 10^0 and 10^2, respectively. The latter have to be understood considering that minimum time step for this simulation is order 10^{-2}. Though

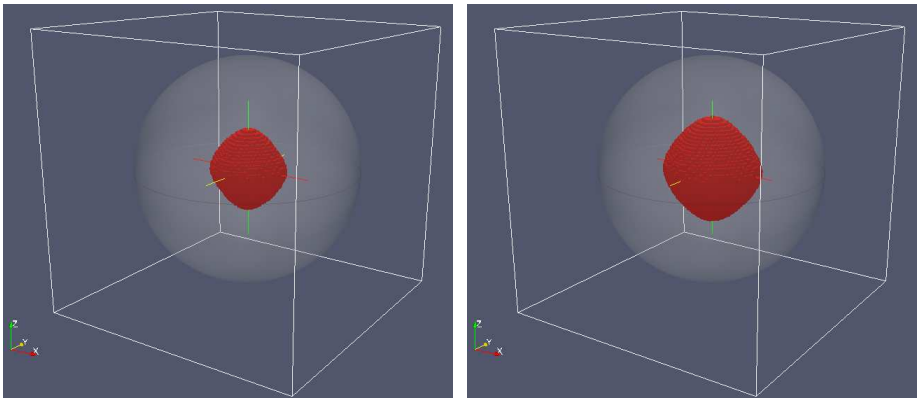
being a heuristic consideration, no mass outflow is observed in simulations performed using this parameters.

The aim of this simulation is to show the evolution of a homogeneous tumor nodule far from the boundary of the domain Ω, in order to give an example of growth not subject to the geometrical constrains imposed by the boundary domain. Conversely, the next setting we will introduce will show the effect on the shape of the nodule because of the heterogeneity nodule composition and the close presence of the boundary of irregular domain.

Figure 5.7 shows the evolution of the nodule shape at four successive times. We plot the contour containing values for the tumor phase $P + Q$ in the range $\left[10^{-6}, 1\right]$. Provided what we have touched on about the approximation of a sphere, the shape of the nodule evolves preserving the isotropic nature of the model and the symmetries of the domain Ω.



(a) T=0.6 (b) T=1.2



(c) T=2.1 (d) T=3.0

Figure 5.7: Time evolution of the nodule shape. (Sphere case).

The internal structure of the nodule evolves in time according to the conservations equations. At $T = 0$ the quiescent cells density is zero all over the computational domain and the proliferative cells start to use oxygen for the mitosis process. This consumption, even if very weak, decreases the value of the hypoxia threshold, being the latter modelled by a smooth function in Figure

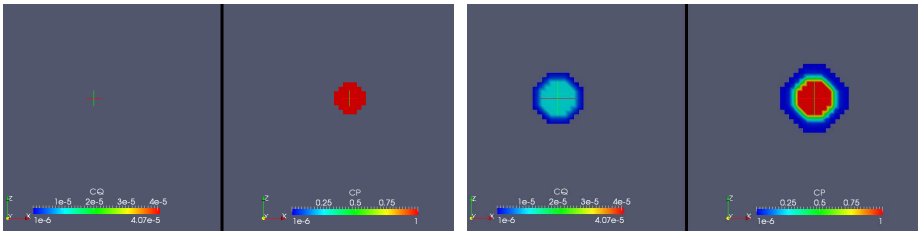
5.5.

Values of γ smaller than 1 switch on the proliferating cells transition to the quiescence state. From this point forward the quiescent phase starts to occupy the core of the nodule and the more the oxygen is consumed more rapidly the proliferating cells change into the Q phase.

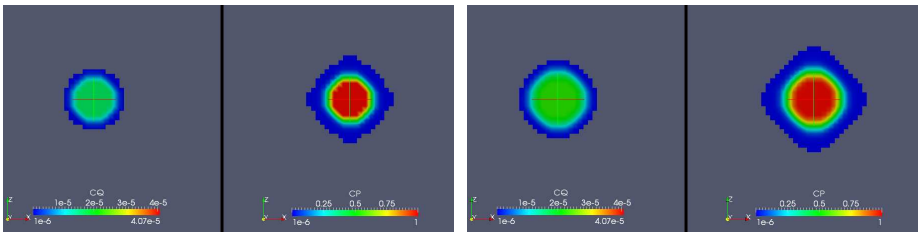
On the other hand the quiescent cells are in a well oxygenated environment, undergoing the opposite transition in their turn. The continuous, but slow, decreasing of available oxygen implies the accumulation of quiescent phase where this consumption is maximum, that is where the density of proliferating phenotype is maximum.

Figure 5.8 accounts for this nodule structure evolution: the two frames in every image have to be considered overlapped and the cross is their shared centre. The transport is responsible for the motion of the cells and the effects of the diffusivity of the numerical schemes used is evident.

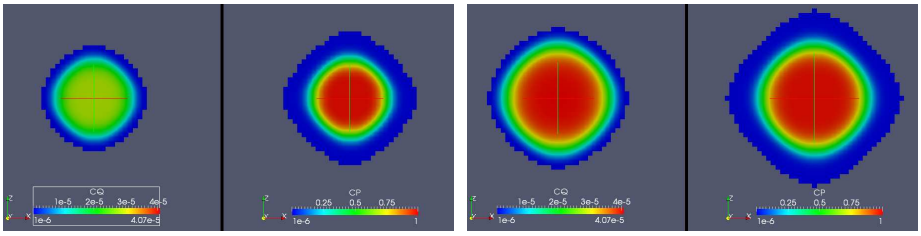
However, even if diffusion is not modelled explicitly, its presence is not so unnatural. Nonetheless, these effects have to be eliminated or at least reduced by the introduction of more appropriate numerical schemes for transport.



(a) T=0 (b) T=0.3



(c) T=0.6 (d) T=1.2



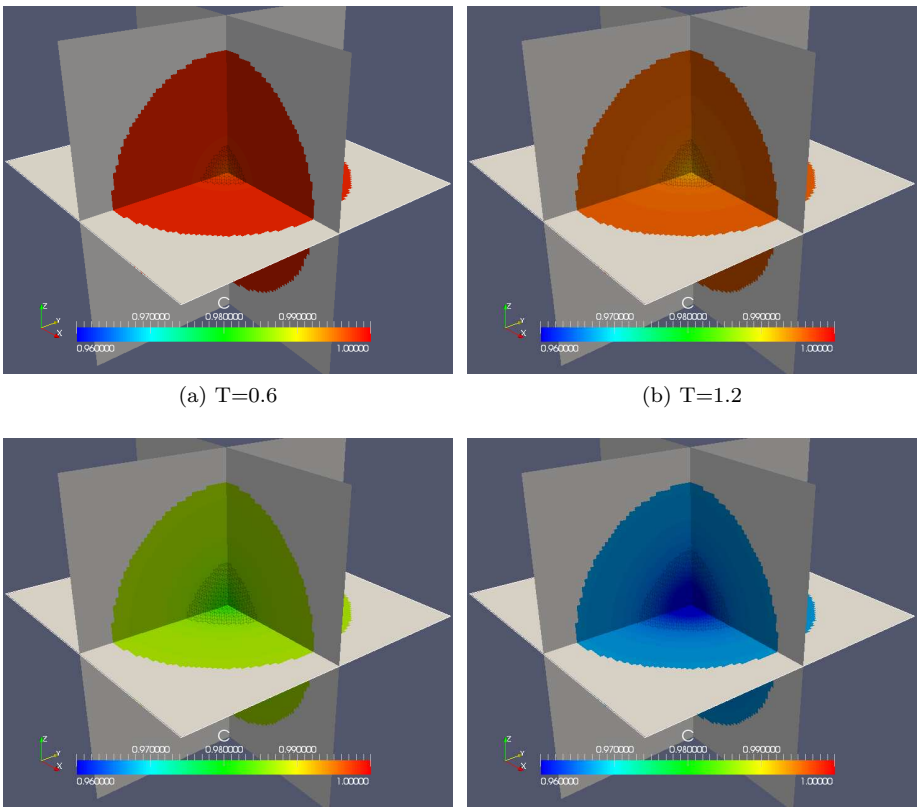
(e) T=2.1 (f) T=3.0

Figure 5.8: Time evolution of the nodule composition. (Sphere case). Slices in the plane $< z, x >$

As fare as the oxygen concentration is concerned, Figure 5.9 shows its time evolution. The presence of the nodule is provided by a black line grid repre-

senting what in Figure 5.7 is given by a red surface. The color scale picks the oxygen consumption out: the nodule plays the role of a sink, using oxygen for the mitosis process, while the diffusion tries to homogenize the oxygen distribution, moving the oxygen from the regions far form the nodule toward the centre of the domain occupied by the tumor. The healthy cells production in not evident, not only because of the action of the diffusion term but also because of the overall high oxygen concentration, keeping the production rate very low.

Outside Ω the value for oxygen concentration is zero for all the simulation. Therefore, no oxygen outflow is observed, proving that at least in this application the numerical method is able to impose homogeneous Neumann boundary conditions to a good level of approximation.



(a) T=0.6 (b) T=1.2

(c) T=2.1 (d) T=3.0

Figure 5.9: Time evolution for oxygen concentration. White color means $C = 0$ and the black lines denote the boundary of the tumor nodule. (Sphere case).

Another consideration needs a little bit more attention: it is not evident in the first three snapshots in Figure 5.9 because the density of the quiescent cells is very low, but in Figure 5.9d we can notice that the core of the nodule is fairly less blue than the region immediately around, meaning an apparently incoherent higher concentration of oxygen. This effect is due to the rising density of cells in the quiescence state and to a proliferation decrease in this area. This phenotype doesn't perform cell division and it consumes less oxygen relative to the proliferating one (a factor 0.01 in the model, see equation (5.24)).

Lung domain with an ellipsoidal nodule. In this section we want to discard some of the symmetries considering in the sphere case. The domain Ω is the lung obtained by segmentation of CT-scans and already introduced in section 5.3.1, exactly in Figure 5.3b.

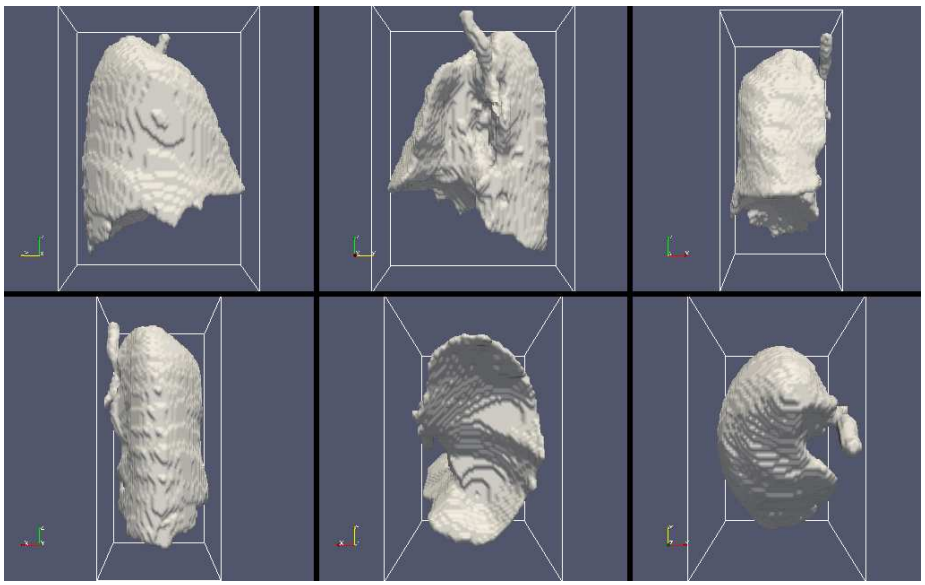
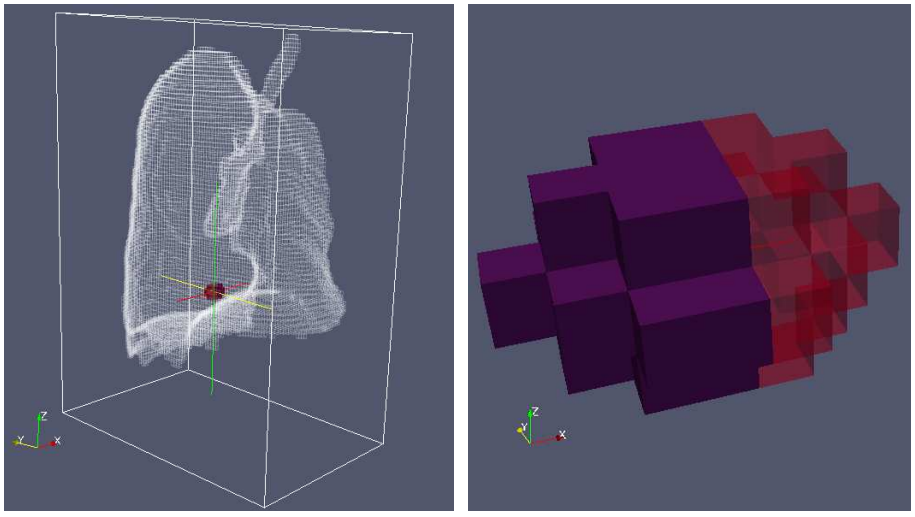


Figure 5.10: Lung sides.



(a) Nodule initial location

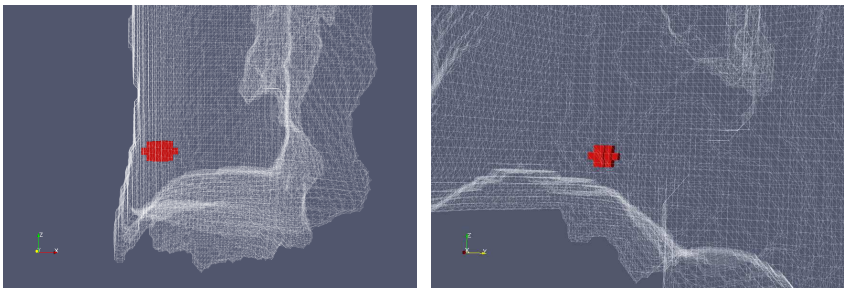
(b) Nodule initial shape, transparent red is for the Q phase

Figure 5.11: Initial geometrical setting

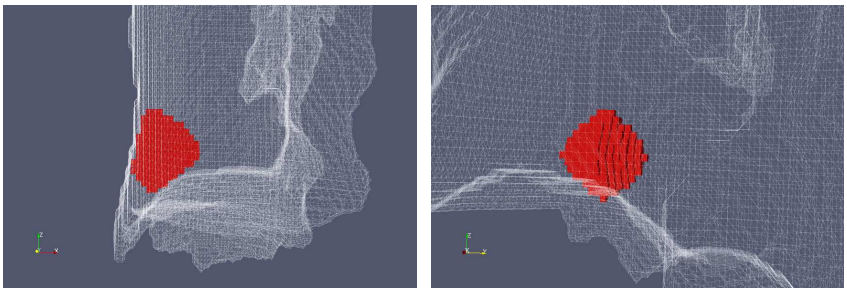
The parameters are unchanged relative to the previous case. Conversely, the nodule is no more a sphere but an ellipsoid. Its initial composition is no more homogeneous as for the sphere, but the ellipsoid is cut in two parts along

the x direction, filling one the part with proliferating cells and the other with quiescent ones. Furthermore, its initial location is no more centred relative to the domain Ω, but it is placed close to the boundary.

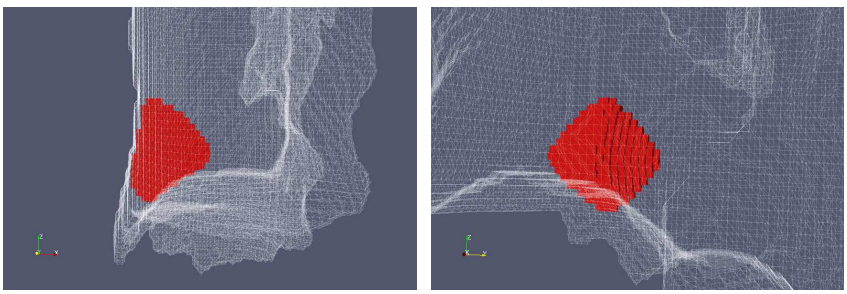
The aim of this simulation is testing the ability of the interface method to impose no mass flow conditions at the boundary of Ω, preventing the tumor phenotypes from going out of the lung. As a consequence we want to show the role of the boundary in determining the shape of growing tumor nodule.



(a) T=0 (b) T=0



(c) T=0.6 (d) T=0.6



(e) T=1.2 (f) T=1.2

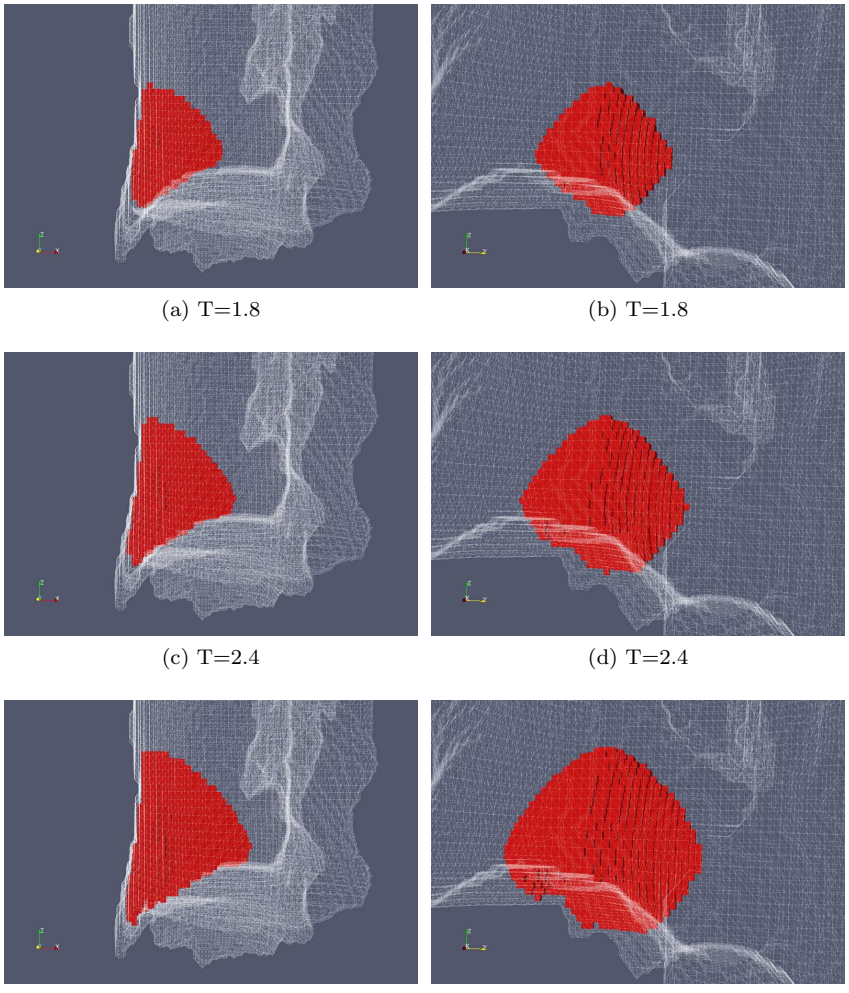
Figure 5.12: Time evolution of the nodule shape. (Lung case, T=0-1.2).).

The domain Ω' is $[0, 0.433] \times [0, 0.697] \times [0, 1]$ and the Cartesian grid is $54 \times 86 \times 100$. The nodule is an ellipsoid centred in $(x_c, y_c, z_c) = (0.082, 0.246, 0.4)$ with semi-axes $a = 0.033, b = 0.025, c = 0.02$. Along the x-axis five-eighths of the nodule are full of the P phase, the remaining the the Q one. Figure 5.11 shows the initial geometries, while Figure 5.10 tries to detail the lung domain.

We want to point out that in this case the initial location of the nodule is very close to the boundary of the domain Ω being a only one grid step far in

the x direction and about three in the other two directions. As a consequence the tumor growth is influenced right from the beginning by the proximity of the boundary.

The nodule increases its size moving toward the inner region of the domain and, following the contour of the domain, it fills the lobes of the lung. The larger its size gets, the more its shape is similar to the shape of the lung. This means that a good approximation of the boundary conditions prevents mass outflow for the phenotypes too.



(a) T=1.8 (b) T=1.8

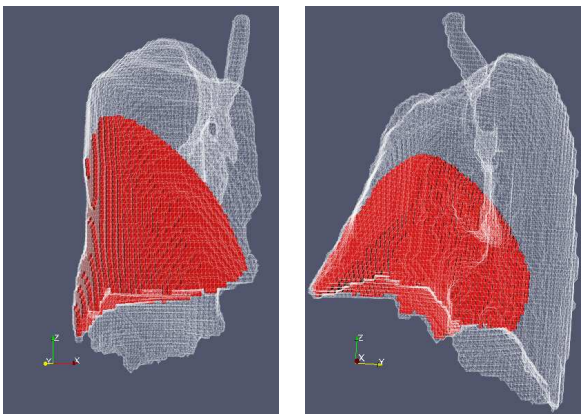
(c) T=2.4 (d) T=2.4

(e) T=3.0 (f) T=3.0

Figure 5.13: Time evolution of the nodule shape. (Lung case, T=1.8-3.0).

The simulation is conducted letting the tumor get exceedingly large, in order to observe the boundary effect. In Figures 5.12 and 5.13 the evolution of the tumor phenotypes is given, magnifying the region occupy by the nodule and offering two points of view to better understand the shape and the approach to the boundary.

We remark that at time $T = 0.6$ the nodule touches the left side of the



(a) T=5 (b) T=5

Figure 5.14: Final time of the evolution of the nodule shape. (Lung case, T=5.0).

boundary and at time $T = 1.2$ the bottom one. At successive times is starts to fill the lower lobe of the lung keeping on growing toward the inner region. At time $T = 5$ in Figure 5.14 the tumor occupies an important portion of the lung. For the sake of precision, we point out that the boundary is the locus of midpoints between couples of grid nodes.

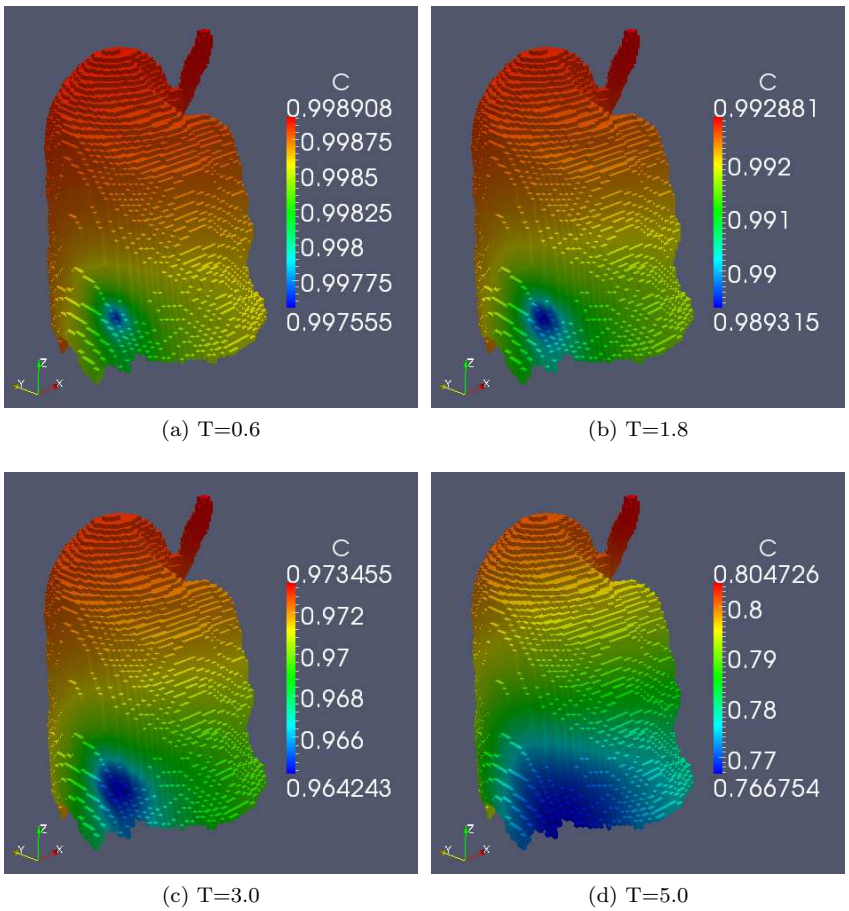
As far as the oxygen consumption is concerned, the main phenomena have been discussed in the sphere case. Oxygen concentration evolves following the same principles, but it attains at the final stage much lower values due to larger tumor mass. Figure 5.15 provides oxygen concentration for four times. Being the nodule so close to the boundary, the boundary itself is a good surface to observe the oxygen consumption. We remark how the lobe in the lower part of the lung plays the role of a trap for the tumor cells, which consume more and more oxygen in a region isolated from the rest of the lung and unable to collect oxygen by diffusion.

We focus now on the composition of the nodule, being the initial structure different from the previous case. The two parts of the nodule are saturated by one of the tumor phenotypes and as for the sphere high concentration of oxygen is available.

Provided the x asymmetry in the nodule shape, it is convenient looking at its structure, extracting a plane containing the axis x and the initial configuration of the phenotypes. We can, therefore, account for the different distribution of P and Q, choosing the plane at $y = y_c$ Figure 5.16 can help showing the two densities and their interplay.

The phases interaction rules have been discussed for the sphere case. Here we want to point out the effect of the initial asymmetric distribution. The proliferating part of the nodule starts dividing, consuming oxygen and moving almost like in the sphere case. Q cells, as for the sphere, appears in the dense core of the proliferating population, as an effect of the oxygen consumption. Finally, the proliferating part of the nodule behaves like the whole nodule in the previous case.

On the other hand, because of the great availability of oxygen, the initial

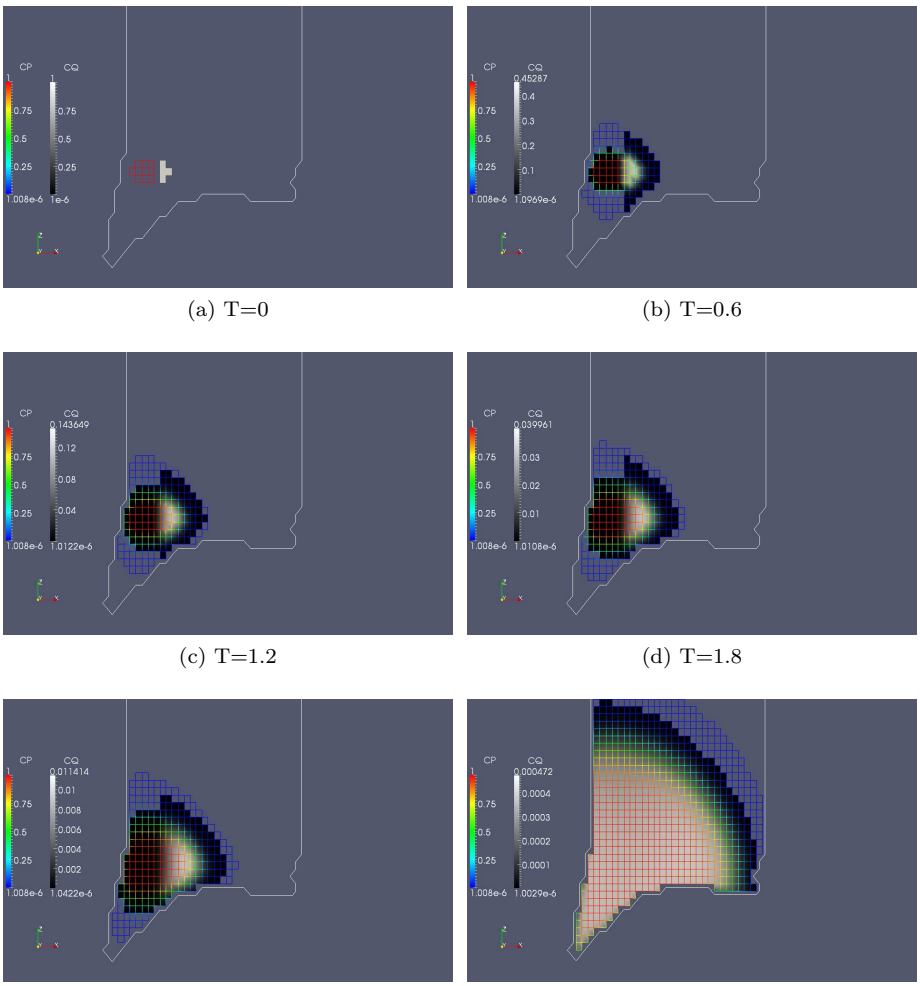


(a) T=0.6 (b) T=1.8

(c) T=3.0 (d) T=5.0

Figure 5.15: Time evolution for oxygen concentration. (Lung case).

core of quiescent cells undergoes a strong reduction in density (see the Q key in figure), starting to be invaded by the proliferating phenotype. In that region especially just after $T = 0$ the conditions are favourable for all the modelled processes producing P cells and most of all for the transition decreasing the Q density. At times $T = 0.6, 1.2, 1.8$ it is evident that the proliferating cells increase mainly in the region initially occupied by the quiescent ones. For this reason, the original quiescent part of the nodule tends to vanish because of this invasion and at time $T = 4.8$ it is no more visible, absorbed by the quiescent cells coming from the P phenotype. At the same time the spatial composition of the nodule has completely lost any evidence of the initial state.



(a) T=0 (b) T=0.6

(c) T=1.2 (d) T=1.8

(e) T=2.4 (f) T=4.8

Figure 5.16: Time evolution of the nodule composition on the plane $y = y_c$. (Lung case).

5.5 Conclusions about the application

In this chapter we have proposed the application of the parallel second order method for elliptic interface problems introduced in chapter 4 to elliptic irregular domain problems in the framework of tumor growth modelling.

In order to solve elliptic problems in irregular domains with homogeneous Neumann boundary conditions with second order accuracy on Cartesian grid we have introduced an approach in the penalty methods spirit. Embedding the original irregular domain in a larger regular one, we redefine the problem as an interface problem with some free parameters, imposing an approximation of original boundary conditions as penalised interface conditions by opportunely choosing the new parameters.

We have provided a two-dimensional example in order to show how the choice of the parameters influences the approximation of the original boundary

conditions, remarking an important warning for the generalized application of the proposed procedure.

The procedure has been applied to elliptic and parabolic problems occurring in the two species Darcy-type model of tumor growth and some results have been shown. These results have provided, at a quality level, a good behaviour of the method, allowing to impose the prescribed boundary conditions. They have shown no mass outflow in both the performed simulations, displaying visible reasons and motivations for further investigations about this use of the interface method.

On the other hand, despite the need for more efforts to reach a closer approximation of the model, treating conservation and numerical diffusion issues with more accurate methods, we believe that the nodule shape and the structure evolution results have shown differences between the two cases, stressing the role of boundaries in tumor growth and motivating the survey in this direction for practical applications.

Chapter 6

Future perspectives

In this final chapter we want to outline some approach to the open questions left by the present work and to introduce possible future developments and new applications.

About the interface method

The second order parallel method for two-dimensional elliptic interface problem, we have introduced in this work, has shown good results relative to the other works in literature for both the error convergence rate and the absolute error. It is based on the introduction of new unknowns located at the interface and a dimension-by-dimension argument.

Using the latter we have proposed here an extension to three-dimensional problems, providing the results of a test conducted on a simple interface which has given the expected second order error convergence and almost the same parallel scaling feature of the two-dimensional version. This preliminary three-dimensional test in the present work make us believe that the extension to three dimension of the method is possible and that it can display the same characteristics in dealing with complex interfaces the method has exhibited in two-dimensional tests. Some evidence has been provided also by the results obtained with the complex lung interface in tumor growth modelling application framework.

Despite all these promises, rigorous test cases have to be performed in order to assert the qualities of the method in solving three-dimensional problems. We aim at proving the second order error convergence rate and reasonable parallel scaling features with test cases allowing comparisons with the methods and a good assortment of test surfaces and problems can be found in [27].

We have referred, introducing the simulations of the tumor growth model, at the lack in explicit treatment of the nodule interface. Still considering tumor growth model, we can think also about organs having discontinuities in some of their properties, brain for instance (grey and white matters). More generally any elliptic interface problem which has to deal with more than an interface would need a development of the computational implementation of the method, if a unique description of all the interfaces is not possible. We want to address this issue in the immediate future.

We have obtained, at least for the tested cases provided in the present work, reassuring performances in parallel scaling tests of the codes, but we think about the absolute value of the performance anyway. For this reason we aim at understanding if and how high performing solvers and preconditioners , like

multigrid, can be used to solve the linear system in the present method and, in that case, what modifications of the method would be needed for their application. In particular, the possible issues coming from the consideration of out-of-grid unknowns have to be investigated.

As far as the elliptic irregular domain problem is concerned, we need to tackle the issue about the control of the normal derivative of the solution on the external side of the original domain. In order to generalize the approach this is the most important question left open in this work. Embedding the irregular domain and rephrasing the boundary problem, we introduce three new entities. In the present work we have chosen to set two of them in the simplest way and to vary the third in order to obtain the searched approximation. However, it may not always be feasible. On the other hand, a generalised approach may rely on the proper consideration of the solution jump and the external source. We need to have further experiences and to investigate more stressing cases, in order to comprehend a general behaviour of the problem and to set a general approach in setting these degrees of freedom.

About the applications of the method

Some of the future developments of the application provided in the present work rely on the improvements of the method itself. Considering more challenging organs, like brain, or modelling the interface of the nodule considering exchanges will be possible when the method will be able to deal with several sub-domains and transmission conditions. Porosity and diffusivity of the model involve jumps between the healthy and the tumor phases, but the order of this jumps is small and they can be properly regularized by the discretization of the elliptic operator. On the other hand, it could be interesting exploring higher values for this jump. Even in this case the method should be able to consider more than one interface, at least the boundary of the original domain and the interface defined by the discontinuity.

We aim to perform other numerical experiences to explore the range of model parameters and simulate different environmental conditions, in order to try the method in several different conditions, collecting more information about its behaviour.

In addition to this, some of the aspects of the numerical framework used to solve the model have to be carried to more accurate standards, like the discretization of transport equations. Nothing new has to be thought, but an important parallel implementation work has to be done.

An important issue we need to handle in future is the validation of model and methods using clinical data. This requires two important procedures on the medical images: segmentation that we have used to get the lung domain and that it is necessary to obtain realistic description of the nodule, and registration. The latter considers the fact that simulations are conducted in a fixed geometrical configuration (organ shape, positions in a reference system), while time series of medical image usually contain several elements varying this configuration (different sensors, patient position, and so on) and it transforms the different images into one coordinate system taken as reference for the numerical experiments.

Finally, other applications of the method will be considered. We have already started to work on a Navier-Stokes solver for incompressible two-phase flows that

uses the present method to solve the Poisson equation for pressure correction. The aim is produce a tool for simulating the dynamics of a sharp interface between two fluids with a large jump in density. Moreover, it will be interesting to couple this solver with rigid body by means of penalisation methods.

Bibliography

[1] The Insight Segmentation and Registration Toolkit. www.itk.org.

[2] The Message Passing Interface Forum Home Page. http://www.mpi-forum.org.

[3] Plateforme Fédérative pour la Recherche en Informatique et Mathématiques en Aquitaine. https://plafrim.bordeaux.inria.fr.

[4] D Ambrosi and L. Preziosi. On the closure of mass balance models for tumor growth. *Math. Models Methods Appl. Sci.*, 12:734–754, 2002.

[5] D. Ambrosi and L. Preziosi. Cell adhesion mechanisms and elasto-viscoplastic mechanics of tumors. *Mech. Model. Mechanobiol.*, 8:397–413, 2009.

[6] D. Ambrosi, L. Preziosi, and C. Verdier. An elasto-visco-plastic model of cell aggregates. *J. Theor. Biol.*, 262:35–47, 2010.

[7] A.R.A. Anderson and M. Chaplain. Continuous and discrete mathematical models of tumor-induced angiogenesis. *Bull. Math. Biol.*, 60:857–900, 1998.

[8] Michael Andreeff, David W Goodrich, and Arthur B Pardee. *Cancer Medicine*, chapter Cell Proliferation, Differentiation, and Apoptosis. Holland-Frey, 5th edition, 2000.

[9] P. Angot, C.-H. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in incompressible flows. *Numer. Math.*, 81(4):497–520, 1999.

[10] P. Araujo and D. L. S. McElwain. A mixture theory for the genesis of residual stresses in growing tissues i: A general formulation. *SIAM J.Appl. Math.*, 65(4):1261–1284, 2005.

[11] E. Arquis and J.P. Caltagirone. Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide-milieux poreux: application la convection naturelle. *C.R. Acad. Sci. Paris II*, 299:1–4, 1984.

[12] I. Babuska. The finite element method for elliptic equations with discontinuous coefficients. *Computing*, 5:207–213, 1970.

[13] Satish Balay, Jed Brown, , Kris Buschelman, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.0.0, Argonne National Laboratory, 2008.

[14] Satish Balay, Jed Brown, Kris Buschelman, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc Web page, 2009. http://www.mcs.anl.gov/petsc.

[15] Satish Balay, William D. Gropp, Lois Curfman McInnes, and Barry F. Smith. Efficient management of parallelism in object oriented numerical software libraries. In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

[16] E.L. Bearer, J.S. Lowengrub, Y.L. Chuang, H.B. Friboes, F. Jin, F.M. Wise, M. Ferrari, D.B. Agus, and V. Cristini. Multiparameter computational modeling of tumor invasion. *Cancer Res.*, 69:4493–4501, 2009.

[17] P. Berthelsen. A decomposed immersed interface method for variable coefficient elliptic equations with non-smooth and discontinuous solutions. *J. Comput. Phys.*, 197:364–386, 2004.

[18] F. Billy, B. Ribba, O. Saut, H. Morre-Trouilhet, T. Colin, D. Bresch, J.P. Boissel, E. Grenier, , and J.P. Flandrois. A pharmacologically based multiscale mathematical model of angiogenesis and its use in investigating the efficacy of a new cancer treatment strategy. *J. Theor. Biol.*, 260:545–562, 2009.

[19] J. Bramble and J. King. A finite element method for interface problems in domains with smooth boundaries and interfaces. *Adv. Comput. Math.*, 6:109–138, 1996.

[20] D. Bresch, T. Colin, E. Grenier, B. Ribba, and O. Saut. Computational modeling of solid tumor growth: the avascular stage. *SIAM Journal on Scientific Computing*, 32(4):2321–2344, 2009.

[21] F. Buret, N. Faure, L. Nicolas, R. Perussel, and C. Poignard. Numerical studies on the effect of electric pulses on an egg-shaped cell with a spherical nucleus. Technical Report 7270, INRIA, 2010.

[22] Jerrold T. Bushberg, J. Anthony Seibert, Edwin M. Leidholdt Jr., and John M. Boone. *The Essential Physics of Medical Imaging.* Lippincott Williams & Wilkins, 2001.

[23] F. Chantalat, C.H. Bruneau, C. Galusinski, and A. Iollo. Level-set, penalization and cartesian meshes : a paradigm for inverse problems and optimal design. *J. Comput. Phys.*, 228:6291–6315, 2009.

[24] M. A. J. Chaplain. Avascular growth, angiogenesis and vascular growth in solid tumours: The mathematical modelling of the stages of tumour development. *Mathl. Comput. Modelling*, 23(6):47–87, 1996.

[25] A. Chauviere, T. Hillen, and L. Preziosi. Modeling cell movement in anisotropic and heterogeneous tissues. *Networks Heterogen. Media*, 2:333–357, 2007.

[26] Z. Chen and J. Zou. Finite element methods and their convergence for elliptic and parabolic interface problems,. *Numer. Math.*, 79:175–202, 1998.

[27] I. Chern and Y.-C. Shu. A coupling interface method for elliptic interface problems. *J. Comput. Phys.*, 225:2138–2174, 2007.

[28] M. Cisternino and L. Weynans. A parallel second order cartesian method for elliptic interface problems. *Comm. in Comput. Phys.*, In Press.

[29] T. Colin, A. Iollo, D. Lombardi, and O. Saut. System identification in tumor growth modeling using semi-empirical eigenfunctions. *M3AS*, 2011. (to appear).

[30] J. Crank and P. Nicolson. A practical method for numerical evaluation of solutions of partial differential equations of the heat conduction type. *Proc. Camb. Phil. Soc.*, 43(3):50–67, 1947.

[31] V. Cristini, X. Li, J.S. Lowengrub, and S.M. Wise. Nonlinear simulations of solid tumor growth using a mixture model: invasion and branching. *J. Math. Biol.*, 58:723–763, 2009.

[32] Shu C.W. and Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.

[33] Shu C.W. and Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes, ii. *J. Comput. Phys.*, 83:32–87, 1989.

[34] A. Deutsch and S. Dormann. *Cellular Automaton Modeling of Biological Pattern Formation*. Birkhäuser, 2005.

[35] A. d'Onofrio. A general framework for modeling tumor-immune system competition and immunotherapy: Mathematical analysis and biomedical inferences. *Physica D*, 208:220–235, 2005.

[36] D. Drasdo and S. Hömme. Individual-based approaches to birth and death in avascular tumors. *Mathematical and Computer Modelling*, 37:1163–1175, 2003.

[37] Olivier Saut et al. eLYSe web page. http://www.math.u-bordeaux1.fr/∼saut/wp/?page_id=201.

[38] R.E. Ewing, Z. Li, T. Lin, and Y. Lin. The immersed finite volume elements methods for the elliptic interface problems. *Mathematics and Computers in Simulation*, 50:63–76, 1999.

[39] R. P. Fedkiw. Coupling an eulerian fluid calculation to a lagrangian solid calculation with the ghost fluid method. *J. Comput. Phys.*, 175:200–224, 2002.

[40] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152:457–492, 1999.

[41] Judah Folkman. *Cancer Medicine*, chapter Tumor Angiogenesis. Holland-Frey, 5th edition, 2000.

[42] M. Garbey and G. Zouridakis. Modeling tumor growth: from differential deformable models to growth prediction of tumors detected in pet images. *Eng. Med. Biol*, 3:2687–2690, 2003.

[43] P. Gerlee and A.R.A. Anderson. An evolutionary hybrid cellular automaton model of solid tumor growth. *J. Theor. Biol.*, 246:583–603, 2007.

[44] F. Gibou, R. P. Fedkiw, L.T. Cheng, and M. Kang. A second order accurate symmetric discretization of the poisson equation on irregular domains. *J. Comput. Phys.*, 176:205–227, 2002.

[45] F. Gibou and R.P. Fedkiw. A fourth order accurate discretization for the laplace and heat equations on arbitrary domains, with applications to the stefan problem. *J. Comput. Phys.*, 202:577–601, 2005.

[46] B. Gompertz. On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. *Phil. Trans. of R.S.*, 115:513–585, 1825.

[47] S. Gottlieb and C.-W. Shu. Total variation diminishing runge-kutta schemes. *Math. Comput.*, 67(221):73–85, 1998.

[48] B. Gustafsson. A fourth order accurate discretization for the laplace and heat equations on arbitrary domains, with applications to the stefan problem. *SIAM Journal of Numerical Analysis*, 39:396–406, 1975.

[49] B. Gustafsson. The convergence rate for difference approximations to general initial boundary value problems. *SIAM Journal of Numerical Analysis*, 18:179–190, 1981.

[50] P. Hahnfeldt, D. Panigrahy, J. Folkman, and L. Hlatky. Tumor development under angiogenic signaling: A dynamical theory of tumor growth, treatment response, and postvascular dormancy. *Cancer Research*, 59:4770–4775, 1999.

[51] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order accurateessentially nonoscillatory schemes, iii. *J. Comput. Phys.*, 71:231, 1987.

[52] J. Huang and J. Zou. A mortar element method for elliptic problems with discontinuous coefficients. *A mortar element method for elliptic problems with discontinuous coefficients*, 22:549–576, 2002.

[53] J.-S. Huh and J.A. Sethian. Exact subgrid interface correction schemes for elliptic interface problems. *Proceedings of the National Academy of Sciences of the United States of America*, 105:9874, 2008.

[54] H. Johansen and P. Colella. A cartesian grid embedded boundary method for poisson's equation on irregular domains. *J. Comput. Phys.*, 147:60–85, 1998.

[55] J.S. Kim, M.A. Stolarska, and H.G. Othmer. A hybrid model for tumor spheroid in vitro: Theoretical development and early results. *Math. Methods Appl. Sci.*, 17:1773–1798, 2007.

[56] A.K. Laird. Dynamics of tumor growth. *Br. J. Cancer*, 18(3):490–502, 1964.

[57] R. J. Leveque and L.Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Numerical Analysis*, 31(4):1019–1044, 1994.

[58] Z.L. Li. A fast iterative algorithm for elliptic interface problems. *SIAM J. Numer. Anal.*, 35:230–254, 1998.

[59] Z.L. Li and K. Ito. Maximum principle preserving schemes for interface problems with discontinuous coefficients. *SIAM J. Sci. Comput.*, 23:339–361, 2001.

[60] Lance A Liotta and Elise C Kohn. *Cancer Medicine*, chapter Invasion and Metastases. Holland-Frey, 5th edition, 2000.

[61] X.-D. Liu, R. P. Fedkiw, and M. Kang. A boundary capturing method for poisson's equation on irregular domains. *J. Comput. Phys.*, 160:151–178, 2000.

[62] Damiano Lombardi. *Problèmes Inverses pour les modèles de croissance tumorale*. PhD thesis, Université Bordeaux 1, 2011.

[63] J. S. Lowengrub, H. B. Frieboes, F. Jin, Y.-L. Chuang, X. Li, P. Macklin, S. M. Wise, and V. Cristini. Nonlinear modelling of cancer: bridging the gap between cells and tumours. *Nonlinearity*, 23:R1–R91, 2010.

[64] P. Macklin and J.S. Lowengrub. An improved geometry-aware curvature discretization for level set methods: Application to tumor growth. *J. Comput. Phys.*, 215:392–401, 2006.

[65] B. Maury. A fat boundary method for the poisson problem in a domain with holes. *Journal of Scientific Computing*, 16:319–339, 2001.

[66] A. Mayo. The fast solution of poisson's and the biharmonic equations on general regions. *SIAM J. Numer. Anal.*, 21:285–299, 1984.

[67] A. Mayo. The rapid evaluation of volume integrals of potential theory on general regions. *J. Comput. Phys.*, 100:236–245, 1992.

[68] A. Mayo and A. Greenbaum. Fast parallel iterative solution of poisson's and the biharmonic equations on irregular regions. *SIAM J. Sci. Stat. Comput.*, 13:101–118, 1992.

[69] P. McCorquodale, P. Collela, and H. Johansen. A cartesian grid embedded boundary method for the heat equation on irregular domains. *J. Comput. Phys.*, 173:620–635, 2001.

[70] A. Bellouquid N. Bellomo and E. De Angelis. The modelling of the immune competition by generalized kinetic (boltzmann) models: Review and research perspectives. *Mathematical and Computer Modelling*, 37:65–86, 2003.

[71] M. Oevermann, C. Scharfenberg, and R. Klein. A sharp interface finite volume method for elliptic equations on cartesian grids. *J. Comput. Phys.*, 228:5184–5206, 2009.

[72] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2003.

[73] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.*, 79(12), 1988.

[74] H. G. Othmer and A. Stevens. Aggregation, blowup and collapse: the abc's of taxis in reinforced random walks. *SIAM J. Appl. Math.*, 57(4):1044–1081, 1997.

[75] K.A. Rejniak. An immersed boundary framework for modelling the growth of individual cells: An application to the early tumour development. *J. Theor. Biol.*, 247:186–204, 2007.

[76] B. Ribba, T. Colin, and S. Schnell. A multiscale mathematical model of cancer, and its use in analizing irradiation therapies. *Theor. Biol. Med.*, 3:7, 2006.

[77] J.B. Ribba, O. Saut, T. Colin, D. Bresch, E. Grenier, and J.P. Boissel. A multiscale mathematical model of avascular tumor growth to investigate the therapeutic benefit of anti-invasive agents. *J. Theor. Biol.*, 243:523–541, 2006.

[78] T. Roose, S.J. Chapman, and P.K. Maini. Mathematical models of avascular tumor growth. *SIAM Review*, 49(2):179–208, 2007.

[79] T. Roose, P.A. Netti, L.L. Munn, Y. Boucher, and R. Jain. Solid stress generated by spheroid growth using a linear poroelastic model. *Microvascular Res.*, 66:204–212, 2003.

[80] Y. Saad. Sparskit a basic tool-kit for sparse matrix computations. http://www-users.cs.umn.edu/ saad/software/SPARSKIT/sparskit.html.

[81] A. Sarthou, S. Vincent, P. Angot, and J.P. Caltagirone. The algebraic immersed interface and boundary method for elliptic equations with discontinuous coefficients. submitted, 2009.

[82] J. A. Sethian. A fast marching level set method for monotonically advancing fronts. *J.A., Proc. Nat. Acad. Sci.*, 93(4):1591–1595, 1996.

[83] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge, UK, 1999.

[84] J. A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*. Cambridge University Press, 1999.

[85] J. A. Sethian. Evolution, implementation, and application of level set and fast marching methods for advancing fronts. *J. Comput. Phys.*, 169:503–555, 2001.

[86] M. Svard and J. Nordstrom. On the order of accuracy for difference approximations of initial-boundary value problems. *J. Comput. Phys.*, 218:333–352, 2006.

[87] K.R. Swanson, C. Bridge, J.D. Murray, and E.C. Alvord Jr. Virtual and real brain tumors: using mathematical modelling to quantify glioma growth and invasion. *J. Neuro. Sci.*, 216:1–10, 2003.

[88] Z. Wang, L. Zhang, J. Sagotsky, and T.S. Deisboeck. Simulating non-small cell lung cancer with a multiscale agent-based model. *Theor. Biol. Med. Model*, 4:50, 2007.

[89] A. Wiegmann and K. Bube. The explicit jump immersed interface method: finite difference method for pdes with piecewise smooth solutions. *SIAM J. Numer. Anal.*, 37(3):827–862, 2000.

[90] Liu X.-D., Osher S., and Chan T. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 126:202–212, 1996.

[91] T.S. Yoo, M. J. Ackerman, W. E. Lorensen, W. Schroeder, V. Chalana, S. Aylward, D. Metaxes, and R. Whitaker. Engineering and algorithm design for an image processing api: A technical report on itk - the insight toolkit. *Proc. of Medicine Meets Virtual Reality - J. Westwood (editor) - IOS Press Amsterdam*, pages 586–592, 2002.

[92] S. a Yu and G.W. Wei. Three-dimensional matched interface and boundary (mib) method for treating geometric singularities. *J. Comput. Phys.*, 227:602–632, 2007.

[93] X. Zhong. A new high-order immersed interface method for solving elliptic equations with embedded interface of discontinuity. *J. Comput. Phys.*, 225:1066–1099, 2007.

[94] Y. C. Zhou and G. W. Wei. On the fictitious-domain and interpolation formulations of the matched interface and boundary (mib) method. *J. Comput. Phys.*, 219:228–246, 2006.

[95] Y. C. Zhou, S. Zhao, M. Feig, and G. W. Wei. High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources. *J. Comput. Phys.*, 213:1–30, 2006.